DEPARTMENT OF
AERONAUTICS AND ASTRONAUTICS
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS 02139

COMPUTATIONAL METHODS FOR AERODYNAMIC DESIGN USING NUMERICAL OPTIMIZATION

by

Martin F. Peeters

CFDL-TR-83-1 February 1983

This research was performed in the Computational Fluid Dynamics Laboratory, MIT, and was supported by NASA Ames Research Center grant NAG 2-115.

Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

COMPUTATIONAL METHODS FOR AERODYNAMIC DESIGN USING NUMERICAL OPTIMIZATION

by

Martin F. Peeters

ABSTRACT

Five methods to increase the computational efficiency of aerodynamic design using numerical optimization, by reducing the computer time required to perform gradient calculations, are examined. Four of these methods have flaws, while one shows promise. The promising method consists of drastically reducing the size of the computational domain on which aerodynamic calculations are made during gradient calculations. Since a gradient calculation requires the solution of the flow about an airfoil whose geometry has been slightly perturbed from a base airfoil, the flow about the base airfoil is used to determine boundary conditions on the reduced computational domain. This method worked well in subcritical flow, but some unresolved problems remain if it is used in supercritical flow.

TABLE OF CONTENTS

LIST OF SYMBOLS

a	speed of sound (m/sec)
c	airfoil chord (m)
C_d	drag coefficient
C_l	lift coefficient
Cp	pressure coefficient
$\mathcal{D}$	doublet strength
f	airfoil geometry
G	constraint function
h	shape function
K	transonic similarity parameter
Ku	curvature = $\frac{d^2(t/\delta)}{dx^2}\left(1+\left[\frac{d(t/\delta)}{dx}\right]^2\right)^{-3/2}$
M	Mach number
OBJ	objective function
r	coefficient of shape function
t	airfoil thickness (m)
u,v	perturbation velocity in x and y direction, respectively
Vol	area within airfoil divided by δ
x,y	Cartesian coordinates
$\bar{x},\bar{y}$	dimensional values of x,y (m)
$\tilde{y}$	transonic lateral coordinate
γ	ratio of specific heats
δ	thickness/chord
ϕ	pertubation velocity potential

Φ velocity potential

Subscripts

b base flow

i,j mesh point indices in x and y directions, respectively

L local condition

ls lower surface

us upper surface

∞ freestream condition

Superscripts

q design iteration number

' pertubation from base flow

CHAPTER 1. INTRODUCTION

Improved methods for the design of airfoils are always a subject of interest in aeronautical engineering. To date the most successful analytical methods for the design of airfoils have relied on some form of inverse calculations. An inverse calculation is one in which the desired flow field is specified and the airfoil shape is solved for, which generates this flow field. Examples of the use of inverse methods can be found in the work of Henne (ref. 1), or Chin and Rizzetta (ref. 2).

Inverse methods can be a definite aide in the design of airfoils but they do have some inherent drawbacks: 1) Inverse methods require apriori knowledge of the desired pressure or velocity distibution along the airfoil, 2) The desired flow field may be impossible to realize with any physically realistic airfoil shape, and 3) Constraints on the airfoils characteristics are not easy to implement.

An alternative approach for the design of airfoils has been proposed by Hicks, Murman, and Vanderplaats (ref. 3). The technique is to desgn airfoils using numerical optimization in which an aerodynamic analysis code is coupled to a numerical optimization code. This method would allow the designer to optimize a single performance characteristic of the airfoil while at the same time constraining other performance characteristics of the airfoil to be within certain values prescibed by the

designer.

This method for designing airfoils is very flexible. It gives the designer freedom to choose which performance characteristics to optimize and how much degradation in other performance characteristics is tolerable. Which performance characteristic is to be optimized and which performance characteristics are to be constraints can be varied, giving the designer accurate information on design tradeoffs.

Initial work with this method has shown that while it seems to work, it requires a considerable amount of c.p.u. time, limiting its usefullness. The objective of this present research is to explore ways in which the computational effciency of designing airfoils using numerical optimization can be increased.

The basic concepts involved in optimization will be reviewed first. Thereafter modifications to the method that could improve the computational efficiency will be discussed. Conclusions and recomendations will then be made.

Optimization Concepts

Consider an airfoil in which the upper surface is defined by the functional relationship f_{us} (x/c) and the lower surface is defined by the functional relationship f_{ls} (x/c). It is desired that a certain performance characteristic of the airfoil is optimized. For example, assume that the drag at zero angle of attack is to be minimzed. In this case the

drag would be called the objective function. In doing the optimization, it is desired that constraints be imposed on other performance characteristics of the airfoil. For example, the airfoil volume is constrained to be greater than a specified minimum and the lift is constrained to be greater than a specified minimum.

To perform the optimization, modifications will have to be made to the airfoil geometry. This is accomplished by adding shape functions to the initial airfoil so the upper surface of the airfoil would be given by

$$f_{us}(x/c) = f_{us_{initial}}(x/c) + r_1 * h_1(x/c) + \ldots + r_n * h_n(x/c)$$

and similarly for the lower surface. The function $h_n(x/c)$ is a shape function. Figure 1 illustrates some examples of shape functions that have been used by Hicks and Vanderplaats (ref. 4) for optimization. The r_n's determine the magnitude of each shape function added to the initial airfoil. These are the only quantities that are varied in the optimization process so the r_n's are refered to as the design variables.

The statement of the problem can be summarized as follows:

Minimize $OBJ(\vec{X})$

Subject to: $G_i(\vec{X}) < 0 \quad i=1,m$

where $\vec{X}$ is a design vector

The following terminology is useful when discussing optimization. The n dimensional space spanned by the vector $\vec{X}$ is refered to as the design space. A constraint is said to be inactive if $G_i(\vec{X}) < 0$; it is said to be violated if $G_i(\vec{X}) > 0$; it is said to be active if $G_i(\vec{X}) = 0$. Since an exact zero is rarely found on a computer, a more reasonable definition for an active constraint is $|G_i(X)| < \epsilon$ where ϵ is a small value. This will be the definition of an active constraint used in this paper. A design is feasible if for all i $G_i(X) < 0$. The minimal feasible design is said to be optimal.

How the optimization procedure actually works is best explained by illustating a simple example. Consider the problem:

Minimize C_d

Subject to :

$C_{l_{min}} - C_l < 0$ (lift constraint)

and

$Vol_{min} - Vol < 0$ (airfoil volume constaint)

where $\vec{X} = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$ r_1 and r_2 are the coefficients of two shape functions

The design space for this hypothetical problem is shown in figure 2. Contours of constant objective function (C_d) and the constraints are illustrated in the design space. Assume that the initial airfoil is given by

$$r_1 = 0$$

$$r_2 = 0$$

and that this initial airfoil is a feasible design.

The optimization process is an iterative procedure in which the following recursive relationship is used

$$\vec{X}^{q+1} = \vec{X}^{q} + \alpha * \vec{S} \qquad (1)$$

where q is the iteration number, vector $\vec{S}$ is the direction of search in the design space and α is a scalar defining the distance of travel in the direction given by $\vec{S}$. Each optimization iteration thus procedes in two steps: First the vector $\vec{S}$ giving the direction of travel is found, then the scalar α is determined.

The procedure for determining $\vec{S}$ is somewhat different depending on whether any constraints are active. In the example of figure 2, it is assumed that no constraints are active initially so the determination of $\vec{S}$ proceeds as follows. Each design variable is separately perturbed to determine its effect on the objective function; thus a finite difference approximation to the gradient of the objective function is constructed as

$$\nabla OBJ \equiv \begin{Bmatrix} \frac{\partial (OBJ)}{\partial x_1} \\ \frac{\partial (OBJ)}{\partial x_2} \\ \vdots \\ \frac{\partial (OBJ)}{\partial x_n} \end{Bmatrix} \cong \begin{Bmatrix} \frac{\Delta (OBJ)}{\Delta x_1} \\ \frac{\Delta (OBJ)}{\Delta x_2} \\ \vdots \\ \frac{\Delta (OBJ)}{\Delta x_n} \end{Bmatrix} \qquad (2)$$

In this example there are only two design variables. For each one an aerodynamic calculation must be performed with the design variable perturbed to determine the change in the objective function.

Whith a finite difference approximation to the gradient of the objective function found, $\vec{S}$ can now be determined. Different optimization schemes use different methods to determine $\vec{S}$. A steepest descent method would just make $\vec{S}$ the negative of the gradient. Conjugate gradient methods (ref. 5) or quasi Newton methods (ref. 6) determine $\vec{S}$ as some function of the gradient. In general, optimization schemes determine $\vec{S}$ as some function of the gradient of the objective function.

With $\vec{S}$ known, α must now be found. α is found by conducting a linear search in the direction of $\vec{S}$ until a minimum is found, or until a constraint becomes active. Again, different optimization codes will use different methods to find α . A typical method would be to perform 3 evaluations of the objective function on the line defined by $\vec{S}$. A quadratic fit is then made with these 3 points and the minimum is found. Similar ideas are used in other methods.

Now equation 2 can be used to determine a new airfoil geometry. In the example given in figure 2 this would cause a movement in the design space from A to B. This procedure is repeated until either a minimum is found in the objective function, or a constraint becomes active. In the example of figure 2 the optimization procedure would move the design to

C where the lift constraint becomes active.

At this point a somewhat different procedure is used to find the vector $\vec{S}$. In addition to finding a finite difference approximation to the gradient of the objective function, a finite difference approximation to the gradient of the constraint function must be found. This is done in the same manner as before. Each design variable is perturbed separately to determine its effect and the objective and constraint functions. The determination of $\vec{S}$ again varies from program to program, but usually the optimization program will try to move the design closer to a minimum and at the same time push it away slightly from the constraint.

Constrained and unconstrained optimization iterations are performed as necessary until a feasible design is found which minimzes the objective function (optimal design). In the example of figure 2, this would move the design to point E. There is no guarantee that this design is at an absolute minimum. There may be many local minima in the design space. In general an improvement in the design will have been made, but to have more assurance in finding the absolute minimum the optimization procedure should be started at different points in the design space. In the example of figure 2 point E is a relative minimum. If the procedure was begun at point F, the absoulute minimum, point G, would have been found.

Examples of Airfoil Optimization

The concepts outlined above have been tested by numerous people. Numerical optimization has been used to optimize low speed, high lift airfoils (ref. 7) ; it has been used to optimize airfoils in transonic flow (ref. 8,9). The results obtained by Hicks, Murman and Vanderplaats (ref. 3) in which airfoils in transonic flow are optimized, will be presented as an example demonstrating the potential that this method has for the design of airfoils.

Their optimization procedure coupled together an aerodynamic analysis code based on the small disturbance transonic potential equation and CONMIN (ref.10), a FORTRAN program for constrained function minimization.

Some results from their work are presented in figure 3. In each case the objective was to minimze drag (the only drag present in this inviscid calculation is wave drag). In each case the freestream Mach number was 0.8; there were seven design variables; the airfoils were symmetric. In the cases of figures 3a and 3b the only constraint was an airfoil volume constraint, the cases of figures 3c and 3d imposed a curvature constraint on the airfoil and a thickness/chord constraint in addition to the airfoil volume constraint. In every case a significant reduction in drag was realized.

Limitations of Airfoil Design Using Numerical Optimization

The above examples indicate that improved airfoil designs can be realized using numerical optimization as a design tool. This method is, however, limited by its excessive appetite for c.p.u. time. The examples presented utilized an aerodynamic analysis code based on the small disturbance transonic equation and had only seven design variables. A more realistic problem would utilize an aerodynamic analysis code based on the full potential equation and might have fifteeen or more design variables. In this case the c.p.u. requirements of numerical optimization would generally be considered too large for the method to be used.

It is found that a significant fraction of the time spent in designing airfoils using numerical optimization is spent on calculating the finite difference approximations to the gradients of the objective and constraint functions. Recall that a gradient calculation is made by separately perturbing each design variable and then performing an aerodynamic analysis to determine the change in the objective function and the active constraint functions. The flow about the unperturbed airfoil provides a good initial guess for the solution of the perturbed airfoil, but computer times are still very large.

The objective of this research is to find a method for computing the gradients of the objective and constraint

functions that require significantly less computer time than the method outlined above. The finite difference approximation to the gradients used only one sided differencing to approximate the gradients. One sided differencing is only first order accurate, suggesting that numerical optimization does not require extremely accurate gradient information. This research will take advantage of this fact and will try to make better use of the fact that the solution for the perturbed body is only slightly different from that of the unperturbed body. It is also an objective of this research to find a method to calculate gradients that is not specific to a particular set of governing equations. That is, the method should be applicable to the small disturbance transonic potential equation, the full potential equation, and Euler's equations.

CHAPTER 2. METHODS FOR FAST GRADIENT CALCULATIONS

Five different methods were examined that could quickly solve for the flow about an airfoil whose geometry has been slightly perturbed. One of these methods shows promise in meeting all the objectives stated above. The five methods are: 1)Perturbation Equations, 2) Local Methods , 3) Local Linearization 4) Method of Integral Relations, 5) Method of Reduced Domains.

The first four methods listed were found to have various shortcomings while the final method listed shows some promise. This section will begin with a brief review of the first four methods listed above, after which a more extensive review of the Method of Reduced Domains will be given.

All preliminary investigations were performed using the small disturbance transonic potential equation. A review of the important details involved in solving this equation is presented in the Appendix.

1)Perturbation Equations:

An obvious first step in solving a flow problem which is a perturbation from a known base flow is to rewrite the governing equations with the parameters split into two parts. The first part would satisfy the base solution and the second part would be a perturbation from the base solution. Writing the potential as

$$\Phi = \Phi_b + \Phi'$$

and plugging this into the small disturbance potential equation yields

$$(K-(\gamma+1)(\Phi_{b_x}+\Phi'_x))(\Phi_{b_{xx}}+\Phi'_{xx}) + \Phi_{b\tilde{y}\tilde{y}} + \Phi'_{\tilde{y}\tilde{y}} = 0 \qquad (3)$$

Discarding higher order terms in the perturbation potential and noting that the base potential satisfies the equation yields the desired perturbation equation

$$(K-(\gamma+1)\Phi_{b_x})\Phi'_{xx} - (\gamma+1)\Phi'_x\Phi_{b_{xx}} + \Phi'_{\tilde{y}\tilde{y}} = 0 \qquad (4)$$

The perturbation form of the potential must also be substituted into the shock jump relations for a complete formulation of the problem. The method was not pursued, however, because equation 4 is essentially no simpler to solve than the original small disturbance equation.

2) Local Methods

Local methods try to relate the pressure at the airfoil surface with the local geometry of the airfoil. This is a method proposed by Davis (ref. 11) in which he uses the work of Spreiter and Alksne and their method of local linearization (ref. 12) as a basis to derive the following equations:

$$C_p = \frac{2}{M_L^2(\gamma+1)}\left\{(M_L^2-1)-[(M_L^2-1)^{3/2} - \tfrac{3}{2}M_L^2(\gamma+1)f_x]^{2/3}\right\} \quad M_L>1 \qquad (5a)$$

$$C_p = -\frac{2}{M_\infty^2(\gamma+1)}\left\{(1-M_\infty^2)-[(1-M_\infty^2)^{3/2} + \tfrac{3}{4}M_\infty^2(\gamma+1)\tfrac{1}{\pi}f_{xx}]^{2/3}\right\} \quad M_L<1 \qquad (5b)$$

where

$$M_\infty^* = \left(1/(1-(\gamma+1)/2\, C_p^*\right)$$

and

$$C_p^* = \frac{2}{\gamma M_\infty^2}\left\{\left(\frac{(\gamma+1)/2}{1+(\gamma-1)/2\, M_\infty^2}\right)^{\frac{\gamma}{1-\gamma}}\right\}$$

To obtain these very simple equations, a considerable number of questionable assumptions had to be made in addition to the original assumptions made by Spreiter and Alksne in their development of local linearization. Becuase of all the assumptions made, the validity of equation 5 was put in doubt so the method was not pursued further.

3) Local Linearization:

An attempt was made to use the work done by Spreiter and Alksne that did not require the plethora of assumptions made in local methods. Local linearization is valid for purely subsonic flow, purely supersonic flow, and flow with free stream Mach number very close to 1. The method will be outlined for purely subsonic flow. The ideas used in this case are also used in the supersonic case and the case where freestream Mach number is near 1.

The analysis begins with the small disturbance transonic potential equation

$$\left(1 - M_\infty^2 - M_\infty^2(\gamma+1)\,\phi_x\right)\phi_{xx} + \phi_{yy} = 0 \qquad (6)$$

Let

$$\lambda = \left(1 - M_\infty^2 - M_\infty^2(\gamma+1)\phi_x\right) \qquad (7)$$

and initially treat λ as a constant. This yields the simple equation

$$\lambda\,\phi_{xx} + \phi_{yy} = 0 \qquad (8)$$

the solution of which is

$$u(x,0) \equiv \Phi_x(x,0) = \frac{1}{(\pi\sqrt{\lambda})} \int_0^1 \frac{df/d\eta}{x-\eta} = \frac{u_i}{\sqrt{\lambda}} \qquad (9)$$

This equation can now be differentiated with respect to x yielding

$$\frac{du}{dx} = \frac{1}{\lambda} \frac{du_i}{dx} \qquad (10)$$

The expression for λ is now substituted back into equation 10 and this ordinary differential equation is solved yielding

$$-2/3k \left(1 - M_\infty^2 - ku\right)^{3/2} = u_i + C \qquad (11)$$

where

$$k = M_\infty^2 (\gamma+1)$$

and C is a constant of integration. The above step is an attempt to compensate for the approximation made initially in the analysis in which λ is treated as a constant. The above steps may seem somewhat arbitrary, but Spreiter and Alksne have shown that this sequence of steps leads to the most reasonable approximation.

The only thing that remains to be done is the evaluation of the constant of integration. The method given by Spreiter and Alksne for the evaluation of C is not used. Equation 11 can be rewritten as

$$C = -\frac{2}{3k} \left(1 - M_\infty^2 - ku\right)^{3/2} - u_i \qquad (12)$$

The flow about a base airfoil is known so substituting in u_{i_b} and u_b in the above equation yields an expression for

C. Note that C will now be a function of x. With C(x) known, u(x,0) can be found from equation 11 yielding

$$u(x,0) = k\left[\left(\frac{3}{2}\left(u_i + C(x)\right)\right)^{2/3} - M_\infty^2 + 1\right] \qquad (13)$$

Equation 13 was used to solve for Cp for the pertu·bed airfoil. The base airfoil used was a parabolic arc with δ = .08. The freestream Mach number was .75. This yielded a flow that was subsonic everywhere. Four perturbations in the form of shape functions were separately added to the base airfoil. The four shape functions are presented in figure 4. Each shape function was multiplied by a factor of .004.

The flow about the perturbed airfoil was solved using equation 13 and also using the finite difference solver outlined in Appendix A. The changes in pressure coefficient from the base to perturbed airfoil for each perturbation is shown in figure 5. The results look very promising. The curves obtained using equation 13 or the finite difference method lie nearly on top of each other. The computer time required to solve the problem with equation 13 is 3 orders of magnitude less than the time required by the finite difference method.

Attempts were made to extend this method to supercritical flow, but these failed. The fundamental problem is that the ideas used in local linearization cannot be applied to mixed flow. The ability to extend this method to the full potential equation or Euler's equations also seems doubtful.

For these reasons, the method was abandonned. If, however, there is a need to design subcritical airfoils where the small disturbance potential equation is a reasonable approximation, then this method should work very well.

4) Method of Integral Relations

The purpose of this method is to reduce the dimension of an equation by 1. How this is done is best explained by illustrating a simple example. Consider the following two dimensional equation

$$\frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \qquad (14)$$

where the boundary conditions

$$G(x,0) = g_1 \quad ; \quad G(x, y_{max}) = g_2$$

are given. This equation can be integrated with respect to y yielding

$$\frac{\partial}{\partial x} \int_0^{Y_{max}} F \, dy + g_2 - g_1 = 0 \qquad (15)$$

If F can be written as a(x)*b(x,y) where b(x,y) is a known function then the integral can be performed yielding

$$\frac{\partial}{\partial x} (B(x) * a(x)) + g_2 - g_1 = 0 \qquad (16)$$

where

$$B(x) = \int_0^{Y_{max}} b(x,y) \, dy$$

This one dimensional equation is much simpler to solve than the original two dimensional equation. The success of

this method depends on the accuracy with which b(x,y) is known. It was hoped that since a solution of a flow that is just slightly perturbed from a base flow is desired, the base flow should provide a reasonable approximation to the function b(x,y).

To use this method on the small disturbance transonic potential equation, the equation must be written in divergence form as follows

$$\partial/\partial x \left(K \phi_x - (\gamma+1)/2 \, \phi_x^2 \right) + \frac{\partial}{\partial \tilde{y}} (\phi_{\tilde{y}}) = 0 \qquad (17)$$

thus in this case

$$F = K \phi_x - (\gamma+1)/2 \, \phi_x^2 \qquad (18a)$$

$$G = \phi_{\tilde{y}} \qquad (18b)$$

$\phi_{\tilde{y}}$ is given at $\tilde{y} = 0$ and goes to 0 as $\tilde{y}$ goes to infinity so the limits of integration are 0 and infinity. Writing F as

$$F(x,\tilde{y}) = F(x,0) * F_b(x,\tilde{y})/F_b(x,0) \qquad (19)$$

gives for the function $b(x,\tilde{Y})$

$$b(x,\tilde{y}) = F_b(x,\tilde{y})/F_b(x,0) \qquad (20)$$

The problem with this is that $F_b(x,0)$ may be equal to zero at some point in the flow. In fact $F_b(x,\tilde{y})$ may be zero at some point in the flow so that with $b(x,\tilde{Y})$ written as a ratio, the possibility always exists that the denominator will be zero.

Other forms for $b(x,\tilde{y})$ were investigated such as assuming that $b(x,\tilde{y})$ was an exponential function or an algebraic function. Again, the flow about the base airfoil

was used as a guide to determine the exact form that $b(x,\tilde{y})$ should take. No reasonable way of using the base airfoil to determine $b(x,\tilde{y})$ was found. For this reason the method could not be used.

5) Method of Reduced Domains

The basic idea behind this method is very simple. When calculating the flow about the perturbed airfoil the same solver is used as the one used to calculate the flow about the base airfoil, but the size of the domain on which the calculation is made is greatly reduced. The motivation behind this is that a perturbation in airfoil geometry will primarily affect the flow very close to the airfoil so that the base flow will provide reasonable boundary conditions for the reduced domain. Figure 6 is a typical comparison of the domain size used to calculate the flow about the base airfoil and the perturbed airfoil.

The reduction in computer time that can be expected using the method of reduced domains comes about not only because the number of mesh points is fewer so the number of calculations per iteration is fewer, but also because fewer mesh points generally lead to a higher convergence rate.

Suppose that a problem is solved using an iterative, finite difference scheme and that we want to reduce the size of the error by 10^{-m}. The number of iterations to do this, p, is given by

$$p \geq m/(\log_{10} \rho) \qquad (21)$$

where ρ is the spectral radius.

For a Laplace equation

$$\phi_{xx} + \phi_{yy} = 0 \qquad (22)$$

solved with a Jacobi iterative scheme on a square domain with Δx and Δy equal, we have

$$\rho = 1 - \pi^2/(N^{1/2}-1)^2 \qquad (23)$$

where N is the number of mesh points.

The rate of convergence is thus dependent on the number of mesh points in this case. While the rate of convergence for the small disturbance potential equation or the full potential equation cannot be found analytically, it is expected that the rate of convergence will be a function of the number of mesh points.

Unlike the first four methods tried this method makes its approximation in the boundary condtions used when solving for the flow about the perturbed airfoil, not in the actual method of solution. Some advantages to this are immediately apparent: 1) Major changes to existing codes are not required since the same solver is used, 2) Application of this method to codes based on different governing equations is possible, 3) A tradeoff exists between accuracy and speed, the larger the domain the more accurate the solution, but also the greater the c.p.u. requirement. This last advantage is important because it gives flexibility in using the method of reduced domains. In some applications the size of the domain could be reduced substatially thereby

greatly reducing c.p.u. requirements. However, the size of the domain could always be increased if accuracy became a problem.

Some trends can be found in the accuracy of using the base flow for boundary conditions by examining the inner and outer expansions used at different Mach numbers to derive the small disturbance potential equation from the full potential equation.

To derive the small disturbance potential equation valid for subsonic and supersonic flow

$$(1-M_\infty^2)\,\phi_{xx} + \phi_{yy} = 0 \qquad (24)$$

from the full potential equation

$$(a^2-\phi_x^2)\,\phi_{xx} + (a^2-\phi_y^2)\,\phi_{yy} - 2\phi_x\phi_y\,\phi_{xy} = 0 \qquad (25)$$

requires the use of inner and outer expansions of the form

$$\phi^i = U_\infty\left[\phi_0^i(x,\hat{y}) + \epsilon_1\,\phi_1^i(x,\hat{y}) + \epsilon_1^2\,\phi_2^i(x,\hat{y}) + \ldots\right] \qquad (26a)$$

$$\phi^o = U_\infty\left[\phi_0^o(x,y) + \epsilon_1\,\phi_1^o(x,y) + \epsilon_1^2\,\phi_2^o(x,y) + \ldots\right] \qquad (26b)$$

where

$$\hat{y} = y/\epsilon_1$$

and ϵ_1 is a small parameter related to the airfoil thickness.

To derive the small disturbance potential equation valid for transonic flow

$$\left[1 - M_\infty^2 - M_\infty^2(\gamma+1)\,\phi_x\right]\phi_{xx} + \phi_{yy} = 0 \qquad (27)$$

from the full potential equation requires the use of the following inner and outer expansions

$$\Phi^{i} = U_\infty \left[\phi_0^{i}(x,y) + \epsilon_2 \phi_1^{i}(x,y) + \epsilon_2^2 \phi_2^{i}(x,y) + \dots \right] \qquad (28a)$$

$$\Phi^{o} = U_\infty \left[\phi_0^{o}(x,\beta) + \epsilon_2 \phi_1^{o}(x,\beta) + \epsilon_2^2 \phi_2^{o}(x,\beta) + \dots \right] \qquad (28b)$$

where

$$\beta = \epsilon_2 y$$

and ϵ_2 is a small parameter related to the airfoil thickness.

The important point is that for supersonic and subsonic flow the inner solution is valid only for y of order ϵ_2, while for transonic flow the solution is valid for y of order 1. This means that a perturbation in airfoil geometry will affect the flow for a greater distance from the airfoil in transonic flow than in subsonic or supersonic flow. The accuracy of using the base flow for boundary conditions at the edge of a given reduced domain should therefore be less in transonic flow than in subsonic or supersonic flow.

A complete outline of the inner and outer expansion procedure can be found in reference 13.

Wind Tunnel Analogy

There is a physical analog to computing the flow about the perturbed airfoil on a reduced domain that offers insight into the problem. Determining the aerodynamic characteristics of an airfoil by performing wind tunnel tests is similar to the method of reduced domains. In each

case the flow is affected by the outer boundary.

The hypothetical wind tunnel in which the perturbed airfoil is tested is, however, special. The solution for the base airfoil is obtained on the full domain so that if the domain is reduced, the exact boundary conditions are availiable at the edge of the reduced domain. If the flow in the interior of the reduced domain is perturbed and then the problem is solved with the same airfoil as before, exactly the same answer as that found on the full domain will be computed. This means that the hypothetical windtunnel has been constructed in such a way as to give aerodynamic characteristics for the base airfoil which are the same as if the base airfoil had been tested in free air. Figure 7 shows the base airfoil in the hypothetical windtunnel. Solving for the flow about a slightly perturbed airfoil is thus like testing the slightly perturbed airfoil in the hypothetical windtunnel.

CHAPTER 3. RESULTS OF STUDIES USING THE METHOD OF REDUCED DOMAINS

Exploratory Studies

Results presented in this section compare the change in Cp caused by perturbations in airfoil geometry as computed on the full domain and the reduced domain. The base airfoil in each case was a parabolic arc at zero angle of attack, and the four perturbations of figure 4 multiplied by a small factor were separately added to the base airfoil.

It is difficult to quantify the performance of the method of reduced domains without doing an optimization run. A subjective assessment of the results will be used initially in this chapter to develop confidence in the method. Later in this chapter an actual optimization test will be performed.

Initially all results presented will be from subcritical tests. Greater difficulty was anticipated for supercritical tests and these are discussed after the subcritical results are properly understood.

In solving the flow on the reduced domain, using as the basis for the solution the small disturbance potential equation, two different types of boundary conditions can be specified. Either the potential can be specified at the edge of the domain (Dirichlet boundary condition) or the normal derivative of the potential, the transverse velocity, can be specified (Neumann boundary condition).

Solving the flow about the perturbed airfoil on the reduced domain and specifying the potential to be that of the base flow on the outer boundary is analogous to testing the perturbed airfoil in the hypothetical windtunnel with a freejet boundary condition at the tunnel walls. That is, the specification of the potential to be that of the base flow on the outer boundary forces the pressure to be that of the base flow on the upper boundary of the reduced domain. In a windtunnel with free jet boundaries, one expects the peak perturbation velocities to be underestimated thus underestimating the peak perturbation Cp.

Figure 8 shows the results where the base potential was specified at the edge of the reduced domain. The flow conditions in this case were:

freestream Mach number = 0.7
thickness/chord = 0.1
perturbation multiplication factor = 0.001

The size of the reduced domain is shown in figure 9. The number of mesh points in this reduced domain was 116 while it was 3965 for the full domain. The underprediction of the change in Cp is consistent with the above arguements. The results are, however , reasonably accurate and the average c.p.u. requirement in computing the flow on the reduced domain was approximentaly 60 times smaller than the c.p.u. requirement for computing the flow on the full domain.

The next boundary condition tested was a Neumann boundary condition on the upper boundary of the reduced domain. The finite difference algorithm used in these

computations required the specification of the potential at the upstream boundary, so the potential at the upstream boundary was specified to be that of the base flow. For convenience, the potential was specified to be that of the base flow at the downstream boundary also. In all computations where a Neumann boundary condition was used at the upper boundary, the above outlined boundary conditions were used at the upstream and downstream boundaries.

Specifying the transverse velocity to be that of the base flow on the edge of the reduced domain is analogous to solid wall boundaries on the hypothetical windtunnel. Solid wall boundaries on a windtunnel constrain the flow to be tangential to the solid wall. In this case one would expect a test in the hypothetical wind tunnel on the perturbed airfoil to overestimate peak velocities thus overestimating the perturbation in Cp.

The results presented in figure 10 are consistent with this prediction. The flow conditions in this case were:

freestream Mach number = 0.7
thickness/chord = 0.1
perturbation multiplication factor = 0.001

The size of the domain used in this case is presented in figure 9. The average c.p.u. requirement for computing the flow on the reduced domain was reduced by a factor of 45 as compared to the c.p.u. requirement of the full domain. The smaller reduction in c.p.u. time in this case as compared to the previous case was anticipated since Neumann boundary conditions generally lead to slower convergence.

Windtunnels usually have ventilated walls to try to lessen the effect of the walls on the flow. The next section describes attempts to correct the boundary conditions used on the edge of the reduced domain in the hope of yieldig a more accurate solution.

Boundary Condition Modifications

Modifications were made to each of the boundary conditions used above. The Dirichlet boundary condition was modified by treating the perturbation in geometry of the airfoil as a perturbation doublet and then adding the perturbation in potential caused by this doublet to the base potential on the outer boundary. The Neumann boundary condition was modified by treating the perturbation in geometry of the airfoil as a wavy wall and using a simplified analysis to determine the effect of this perturbation on the transverse velocity at the boundary.

In Appendix A it is shown that in the far field the airfoil is treated as though it were a doublet. This treatment of the airfoil allows the potential to be determined in the far field by use of equation A5. This equation is accurate only if it is used at points a considerable distance from the airfoil (at least 1 chord length).The accuracy of equation A5 diminishes when it is applied closer and closer to the airfoil, but how rapidly the accuracy decays is not known. As an approximation, this equation was used to determine the change in potential at

the edge of the reduced domain caused by the perturbation in airfoil geometry.

The doublet was positioned on the airfoil at the chord station where the amplitude of the perturbation was at a maximum. Equation A6 is the equation for the doublet strength. The equation has one part due to the airfoil volume and a second nonlinear part. The contribution from the nonlinear part is generally small so in computing the perturbation doublet strength it was negleted. The equation for the doublet strength was therefore

$$D = 2 * \int_0^1 f'(\xi)\, d\xi \qquad (29)$$

where $f'(\xi)$ is the perturbation in airfoil geometery.
Figure 11 presents the results obtained with the above outlined boundary conditions. The flow conditions for this test were:

freestream Mach number = 0.7
thickness/chord = 0.1
perturbation multiplication factor = 0.001

The size of the reduced domain is given in figure 9. The two curves lie reasonably close to each other. The average computing time required on the reduced domain was approximately 40 times less than that required on the full domain.

Modifications of Neumann boundary conditions is similar to changing the hypothetical windtunnel shape. The problem is to determine the extent of the modifications required. This problem can be restated as the problem of determining

how much of the perturbation in transverse velocity at the airfoil surface is translated to the edge of the reduced domain. As a guideline to determine the decay in the transverse velocity a simple wavy wall model for the perturbation is used.

Consider the small disturbance transonic potential equation

$$(K-(\gamma+1)\phi_x)\phi_{xx} + \phi_{\tilde{y}\tilde{y}} = 0 \qquad (30)$$

subject to the boundary conditions

$$\phi_x \; ; \; \phi_{\tilde{y}} \quad \text{are finite at } \infty$$

and

$$V(x,0) \equiv \phi_{\tilde{y}}(x,0) = \theta \cos\theta x$$

Let us solve this equation for subsonic flow where $(K-(\gamma+1)\phi_x)$ is always greater than zero. To make the mathematics tractable let us approximate the coefficient of the ϕ_{xx} term as a constant, λ^2. This equation, with the boundary conditions given, can be solved in closed form using separation of variables. The solution is

$$\phi_{\tilde{y}}(x,\tilde{y}) = \theta(\cos\theta x)\, e^{-\tilde{y}\theta\lambda(x,0)} \qquad (31)$$

Note that the local value of λ is used at every point.

The important information from equation 31 is the rate at which velocities at Y = 0 decay. Using the decay rate of this equation gives for the velocity at any point Y

$$\phi_{\tilde{y}}(x,\tilde{y}) = \phi_{\tilde{y}}(x,0)\, e^{-Y\theta\lambda(x,0)} \qquad (32)$$

Equation 32 can be used to approximate the perturbation velocity at the edge of the reduced domain. To use equation 32 , θ , which determines the wavelength of the wavy wall must be known. Since the airfoil is similar to a half sine wave, θ was made to be π .

Figure 12 presents the results of using the approximations just oulined. The flow conditions in this test were:

freestream Mach number = 0.7
thickness/chord = 0.1
perturbation multiplication factor = 0.001

The size of the reduced domain is given by figure 9. The two curves lie very nearly on top of each other, indicating that the above approximation is a very good one. The average computing time required on the reduced domain was 50 times less than that required on the full domain.

These subcritical results are encouraging. The results obtained with the four different boundary conditions used above appear to be at least reasonably accurate and sometimes very accurate. The saving in c.p.u. time in each case was substantial.

The same four boundary conditions outlined above were tested in supercritical flow. The one difference is that the wavy wall formula given above is valid only in subsonic flow. A supersonic wavy wall formula is required for cases where the flow at the edge of the outer boundary is supersonic. The derivation of this formula is completely analogous to the derivation presented for the subsonic flow.

The result is

$$\phi_{\tilde{y}}(x,\tilde{y}) = \phi_{\tilde{y}}(x,o)\left[x - \lambda\,\tilde{y}\right] \qquad (33)$$

As can be seen, the transverse velocity at the airfoil surface does not decay but propogates along characteristics. In locally supersonic flow the boundary condition was therefore applied as follows: The slope of the characteristic was determined from the local Mach number at the boundary. The characteristic was then approximated as a straight line. This line was followed to the airfoil surface where the perturbation transverse velocity was known. This transverse velocity was then propogated back up along the approximation of the characteristic to the boundary of the reduced domain, where it was added to the base transverse velocity. This procedure is illustrated in figure 13.

The runs made with the boundary conditions being: 1) The base potential, 2) The base transverse velocity, and 3) The base potential with the potential due to a perturbation doublet added to it, were all run under the same conditions:

freeesteam Mach number = 0.82
thickness/chord = 0.1
perturbation multiplication factor = 0.001

The size of the domain used is shown in figure 9.

Figure 14 shows the results where the base potential was specified on the edge of the reduced domain. The effect of the perturbation doublet was negligible , the results being essentially those presented in figure 14. The results in these cases are not very good. The largest changes in Cp

occur near the shock and these are not captured well. When the base transverse velocity was specified at the upper boundary of the reduced domain, the computations would not converge to a solution.

The tests in subcritical flow using the wavy wall approximation yielded the best results so it was hoped that they would yield good results in the supercritical case. The initial test was a conservative one in which the flow was only slightly supercritical. The test conditions in this case were:

freestream Mach number = 0.78
thickness/chord = 0.10
perturbation multiplication factor = 0.0014

The size of the reduced domain used is shown in figure 9.

Results are presented in figure 15. As can be seen the results are not very accurate. It seems that supercritical flow is much more sensitive to boundary conditions than subcritical flow.

The flexibility of the reduced domain method was put to use to try to overcome these difficulties. The outer edge of the domains used generally crossed through a region with supersonic flow. This was the probable cause of dissappointing results just presented. To get around this, the shape of the domain was altered so that the outer boundary was always in subsonic flow. A typical domain for supercritical flow calculations is presented in figure 16. Figure 16 also shows the boundary conditions tested. The base potential was specified everywhere except at the very

top of the "notch" part of the domain. There, either the base potential was specified or the base transverse velocity was specified.

For each boundary condition a series of tests were conducted in which the domain size was increased until consistently acceptable results were obtained. For these tests the flow conditions were:

freestream Mach number = 0.82
thickness/chord = 0.10
perturbation multiplicatin factor = 0.001

Figure 17 presents what are considered acceptable results for the Dirichlet boundary condition and figure 18 shows results for the Neumann boundary condition. The corresponding domain size required to achieve these results is shown in figure 19. The number of mesh points in the reduced domain with Neumann boundary conditions was 218 while the number of mesh points in the reduced domain with Dirichlet boundary conditions was 1140. Recall that the the number of mesh points in the full domain was 3965. As can be seen, the size of the domain required to obtain reasonable results using the Dirichlet boundary condition is considerably larger than the domain required to achieve reasonable results using the Neumann boundary condition. An explanation for this phenomena has not yet been found.

The average c.p.u. time required to get the results on the reduced domain is about 30 times less than on the full domain if a Neumann boundary condition is used and about 6 times less if a Dirichlet boundary condition is used. While

the reduction in c.p.u. time is not as large as the reduction realized in subcritical flow, it is still substantial.

Summary of Results of Exploratory Studies

The results can be summarized as follows:

Subcritical Flow

1) Reasonably accurate results can be obtained on a substantially reduced domain.

2) There is flexibility in applying boundary conditions. Different boundary conditions yield different results but all of these results seem acceptable.

3) The c.p.u. requirement of the reduced domain calculations is from 40 to 60 times lower than the c.p.u. requirement for full domain calculations, depending on boundary conditions applied.

Supercritical Flow

1) Reasonably accurate results can be obtained on a substantially reduced domain but the outer boundary of the reduced domain must be in subsonic flow.

2) There is no flexibility in applying boundary conditions. Fast and reliable results can only be obtained with the use of Neumann boundary conditions.

3) The c.p.u. requirement of the reduced domain calculations is 30 times smaller than the c.p.u requirement of full domain calculations.

Note that the savings in computer time using the method of reduced domains came about primarily because of the reduction in the number of calculations required per iteration. The reduction in the actual number of iterations on the reduced domain was not large, always being less than a factor of 2.

Full Potential Equation Calculations

The knowledge gained from applying the method of reduced domains to an aerodynamic analysis code based on the small disturbance potential equation is now applied to the full potential equation. The full potential equation is considerably more accurate than the small disturbance potential equation and so is used much more frequently in the design of airfoils.

A computer code that solves the full potential equation is much more complex than one that solves the small disturbance potential equation. FL06, a nonconservative full potential equation solver written by Antony Jameson, was used in these tests. FL06 was chosen because it is the aerodynamic analysis code in use at NASA Ames for numerical optimization of airfoils. A complete outline of the fundamentals involved in solving the full potential equation can be found in reference 14.

Tests were conducted using FL06 in which the size of the computational domain was greatly reduced. FL06 maps the airfoil onto a unit circle and then performs an inversion such that infinity is mapped into the origin of the unit circle. Thus the computational domain used in FL06 is the inside of a unit circle . The outer boundary of the reduced domain must be in subsonic flow, so for a typical case of a lifting airfoil with a supersonic zone on the upper surface of the airfoil, the computational domain would be like that shown in figure 20. Because of the coordinate system used in FL06, a reduction in domain size like the reduction shown in figure 20, reduced the number of mesh points by a factor of only 5 or 6.

FL06 also uses two steps in solving the equation: a relaxation step, and a fast solver step. The relaxation step gives slow convergence, but can be used on irregular domain shapes and is valid for supersonic flow. The fast solver step gives very rapid convergence in subsonic flow, but becomes less and less stable as the local Mach number increases and goes unstable in supersonic flow. Also, the fast solver can only be used on regular domain shapes. The two steps combined converge reasonably fast in transonic flow.

Because the reduced computational domain is irregular in shape, the fast solver cannot be used over the whole domain. Therefore, the fast solver is used only in the region shown in figure 20. The relaxtion step is used over the whole

domain. Not being able to use the fast solver on the complete reduced domain does not slow convergence since that part of the domain where it is not used is a region of supersonic flow.

Tests were run on the airfoil shown in figure 21 at zero angle of attack, with a freestream Mach number of 0.8 resulting in a supersonic zone on the upper and lower surface of the airfoil. A single perturbation given by

$$Pert = 0.001 * \sin(\pi x) \qquad (34)$$

was added to the upper and lower surface of the airfoil. The boundary conditions used at the edge of the reduced domain were: 1) Potential specified to be that of the base flow everywhere, 2) Potential specified to be that of the base flow everywhere except on the "notch" part of the grid where the normal velocity was specified to be that of the base flow. An unexpected result was found from these tests. While in case 1 the solution did not capture the shock movement as expected from tests performed on the small disturbance potential solver, case 2 converged so slowly that the required c.p.u. time was as large as that needed to solve for the perturbed flow on the full domain.

This fundamental difference in results from the small disturbance potential solver and FL06 did not seem reasonable. One important difference in the two codes was that FL06 is a nonconservative code while the small disturbance solver is a conservative code. A comparison was

made on the rate of convergence of the conservative and nonconservative forms of the small disturbance code. These tests showed that while the two forms of the code had nearly identical convergence rates on the full domain, the conservative form of the code converged in fewer than half the number of iterations required by the nonconservative form of the code on the reduced domain. While not a conclusive result, this strongly suggests that the method of reduced domains can be made to work for supercritical flow if the aerodynamic analysis code is conservative.

Optimization Calculation

Since the method of reduced domains could not be made to work in supercritical flow using FL06, the method was tried on a subcritical case. The objective of this test was not to design a better airfoil but was to demonstrate that the method of reduced domains will produce reasonable results in an actual optimization test.

The optimization code used for this test is called QNMDIFF. It employs a quasi -Newton method for unconstrained optimization. The base airfoil used for this test was a 17 percent thick airfoil designed for general aviation applications (GA(W)-1). The airfoil geometry can be found in reference 15. Computations were made at a Mach number of 0.22 and at zero angle of attack. The objective function used was the sum of the differences between a target pressure distribution along the upper surface of the airfoil

and the actual pressure distribution, that is

$$OBJ = \sum_{i=0}^{n} \left(C_{p_i} - C_{p_{i_{Targ}}} \right)^2 \qquad (35)$$

The target pressure distrabution used was that of the actual (GA(W)-1) airfoil. The initial airfoil used was the (GA(W)-1) airfoil to which four shape functions had been added. Thus the optimized airfoil would be one in which the magnitude of the four shape functions was 0. The four shape functions used were the ones shown in figure 4. The initial magnitudes of the shape functions were:

function 1 magnitude = 0.0100
function 2 magnitude = 0.0087
function 3 magnitude = 0.0093
function 4 magnitude = 0.0021

Table 1 compares the design vector and value of the objective function at the end of each optimization iteration as computed in the regular way and using the method of reduced domains. Figure 22 presents graphically the objective function history of the two methods. In each case the objective function had been reduced substantially after 6 iterations and the design vector was very close to the exact answer of 0,0,0,0.

This optimization test required 40 aerodynamic evaluations in gradient calculations and 21 aerodynamic evaluations in line search calculations. QNMDIFF sometimes requires central differences in gradient calculations which is why the number of aerodynamic evaluations for gradient calculations is not 4*6=24. Each gradient calculation took

approximately one tenth the c.p.u. time on the reduced domain as compared to the full domain. The total amount of computer time required for this optimization calculation using the method of reduced domains was aproximately 45% of the time required in using the standard method.

CHAPTER 4. CONCLUSIONS AND RECOMMENDATIONS

The method of reduced domains has the potential to significantly reduce the computer time required in designing airfoils using numerical optimization. There are, however, unresolved complications in using the method if the flow is supercritical. The use of a nonconservative potential code leads to unacceptably slow convergence rates when computations are performed on the reduced domain. It is probable that this problem would be eliminated if a conservative potential code was used.

Future work on the method of reduced domains should concentrate on understanding the effect that different boundary conditions have on computations using a reduced domain. This paper has presented results of using different boundary conditions. Explaining why one boundary condition works better than another has proved elusive. With the proper understanding of the effect of boundary conditions, the full potential of this method would be known.

This research has concentrated on reducing the computer time required to determine the gradients needed by the optimization program. Research is also warranted on increasing the efficiency of the line search. Making full use of the fact that the line search is a one dimensional problem could yield a signifigant reduction in c.p.u. requirements. Another area of possible research is in determining the best way to modify the airfoils shape.

Different design variables will yield different design spaces and one design space may be much more conducive to numerical optimization than another.

REFERENCES AND BIBLIOGRAPHY

1) Henne, P.A., "An Inverse Transonic Wing Design Method", AIAA Paper 80 - 0330, January 1980

2) Chin, W.C. and Rizzetta, D.P., "Airfoil Design in Subcritical and Supercritical Flows", Journal of Applied Mechanics, Vol 46, No 4, December 1979, pp 761-766

3) Hicks, R.M.; Murman, E.M. and Vanderplaats, G.N., " An Assesment of Airfoil Design by Numerical Optimization ", NASA TM X 3092, July 1974

4) Hicks, R.M. and Vanderplaats, G.N., "Application of Numerical Optimization to the Design of Supercritical Airfoils Without Drag Creep", Society of Automotive Engineers Business Aircraft Meeting Century 11, Witchita Kansas, March 29 - April 1 1977, 770440

5) Fletcher, R. and Reeves, C.M., "Functional Minimization by Conjugate Gradients", Computer J., Vol 7, No 2, July 1964, pp 149-154

6)Broyden, C.G., "Quasi - Newton Methods and Their Application to Function Minimization", Math. of Comp., Vol 21, No 99, 1967, pp 368-380

7) Hicks, R.M. and Vanderplaats, G.N., "Application of Numerical Optimization to the Design of Low-Speed Airfoils", NASA TM X 3213, March 1975

8) Aidala, P.V., "Transonic Fighter Design Using Numerical Optimization", ICAS Paper No 80-2.1, October 1980

9) Haney, H.P.; Johnson, R.R. and Hicks R.M., "Computational Optimization and Windtunnel Tests of Transonic Wing Designs", AIAA paper 79-0080, January 1979

10) Vanderplaats,G.N., "CONMIN, A FORTRAN Program for Constrained Function Minimization", NASA TM X 62,282, 1973

11) Davis, W.H., "Technique for Developing Design Tools From the Analysis Methods of Computational Aerodynamics", AIAA paper 79-1529, July 1979

12) Spreiter, J.R. and Alksne, A.Y., "Thin Airfoil Theory Based on the Approximate Solution of the Transonic Flow Equation", NACA Report 1359, 1958

13) Ashley, H. and Landahl, M. , Aerodynamics of Wings and Bodies, Addison - Wesley Publishing Company Inc,. 1965

14) Jameson, A., "Transonic Flow Calculations", Numerical Methods in Fluid Dynamics, Wirz, H.J. and Smolderen, J.J., Editors, Hemisphere Publishing Corporation, 1978

15) McGhee, R.J. and Beasley, W.D., "low Speed Aerodynamic Characteristics of a 17 Percent Thick Airfoil Section Design for General Aviation Applications" , NASA TN D 7428, December 1973

16) Murman, E.M. and Cole, J.D., "Calculation of Plane Steady Transonic Flows", AIAA Journal, Vol 9, No 1, January 1971, pp 114-121

17) Murman, E.M., "Analysis of Embedded Shock Waves Calculated by Relaxation Methods", AIAA Journal, Vol 12, No 5, May 1974

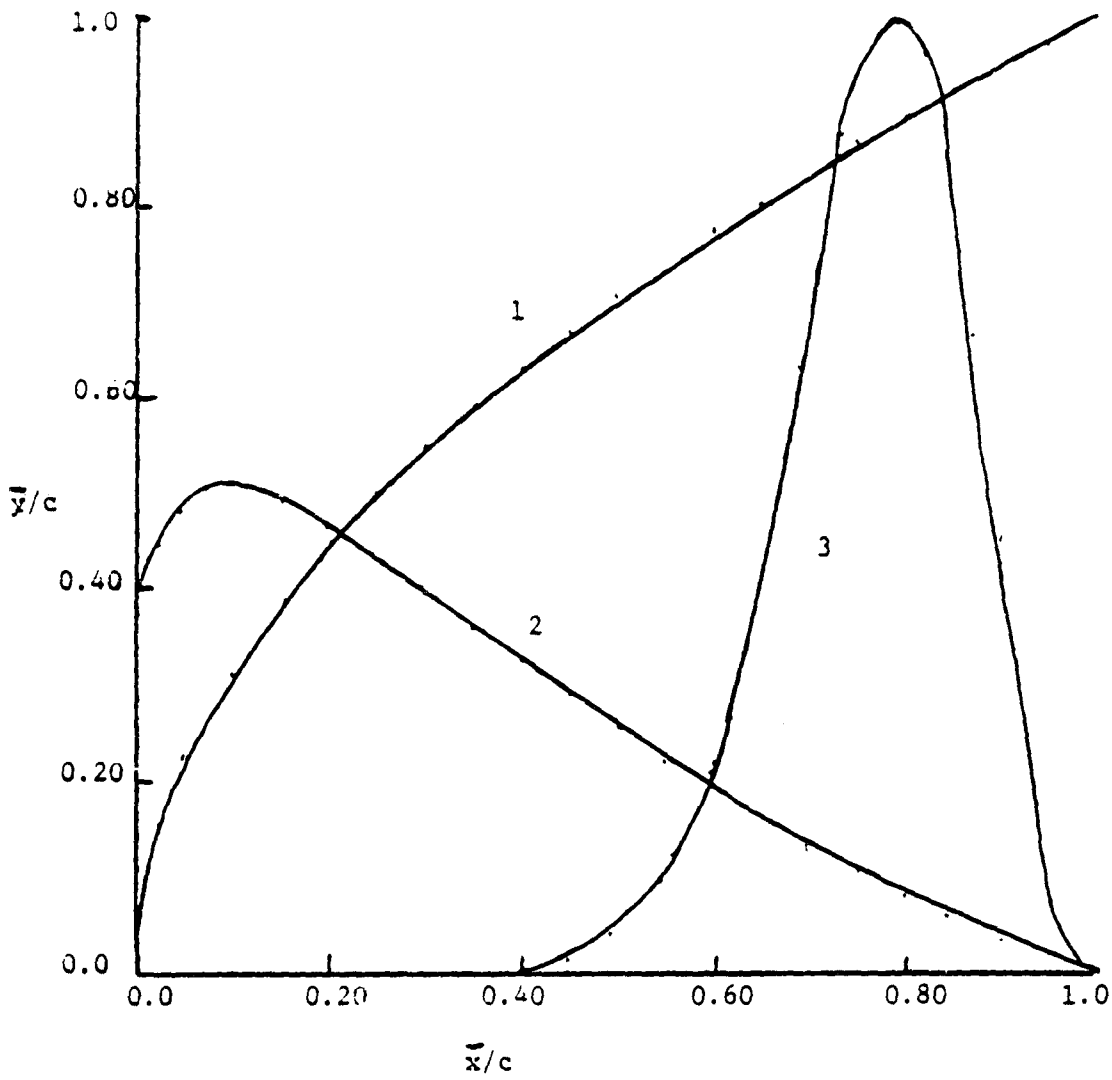
18) Isaacson, E. and Keller, H.B., Analysis of Numerical Methods, John Wiley and Sons Inc., 1966

19) Liepmann, H.W. and Roshko, A. , Elements of Gas Dynamics, John Wiley and Sons Inc., 1963

20) Smith, G.D., Numerical Solution of Partial Differential Equations: Finite Difference Methods, Claredon Press, 1979

Regular Method			Reduced Domain Method		
Iter	Design Vector * 10^2	OBJ	Iter	Design Vector * 10^2	OBJ
0	1.000, 0.870 0.930, 0.210	0.4651	0	1.000, 0.870 0.930, 0.210	0.4651
1	0.0788, -0.0697 0.4923, 0.3342	0.0399	1	0.680, -0.0786 0.471, 0.299	0.0349
2	0.0886, -0.1634 0.1030, -0.0254	0.00234	2	0.119, -0.125 0.115, -0.0279	0.00247
3	0.0886, -0.0865 0.0866, -0.0380	0.00146	3	0.0750, -0.143 0.0737, -0.0433	0.00182
4	0.00245, -0.0146 0.0258, -0.0259	$.255*10^{-3}$	4	-0.0100, -0.0569 0.0316, -0.0266	$.653*10^{-3}$
5	0.00645, -0.00107 0.00913, 0.00482	$.101*10^{-3}$	5	0.00375, 0.00277 0.0156, -0.0205	$.218*10^{-3}$
6	0.00845, -0.00435 0.000612, 0.00142	$.878*10^{-4}$	6	0.00790, -0.0175 0.0102, -0.0117	$.138*10^{-3}$

Table 1. - Details of Optimization Test



1) $\bar{y}/c = \sqrt{\bar{x}/c}$

2) $\bar{y}/c = (\bar{x}/c)^{.2}\left[1-(\bar{x}/c)\right]/e^{(\bar{x}/c)}$

3) $\bar{y}/c = \left(\sin\left(\pi(\bar{x}/c)^{3.10(}\right)\right)^3$

Figure 1. - Examples of Shape Functions

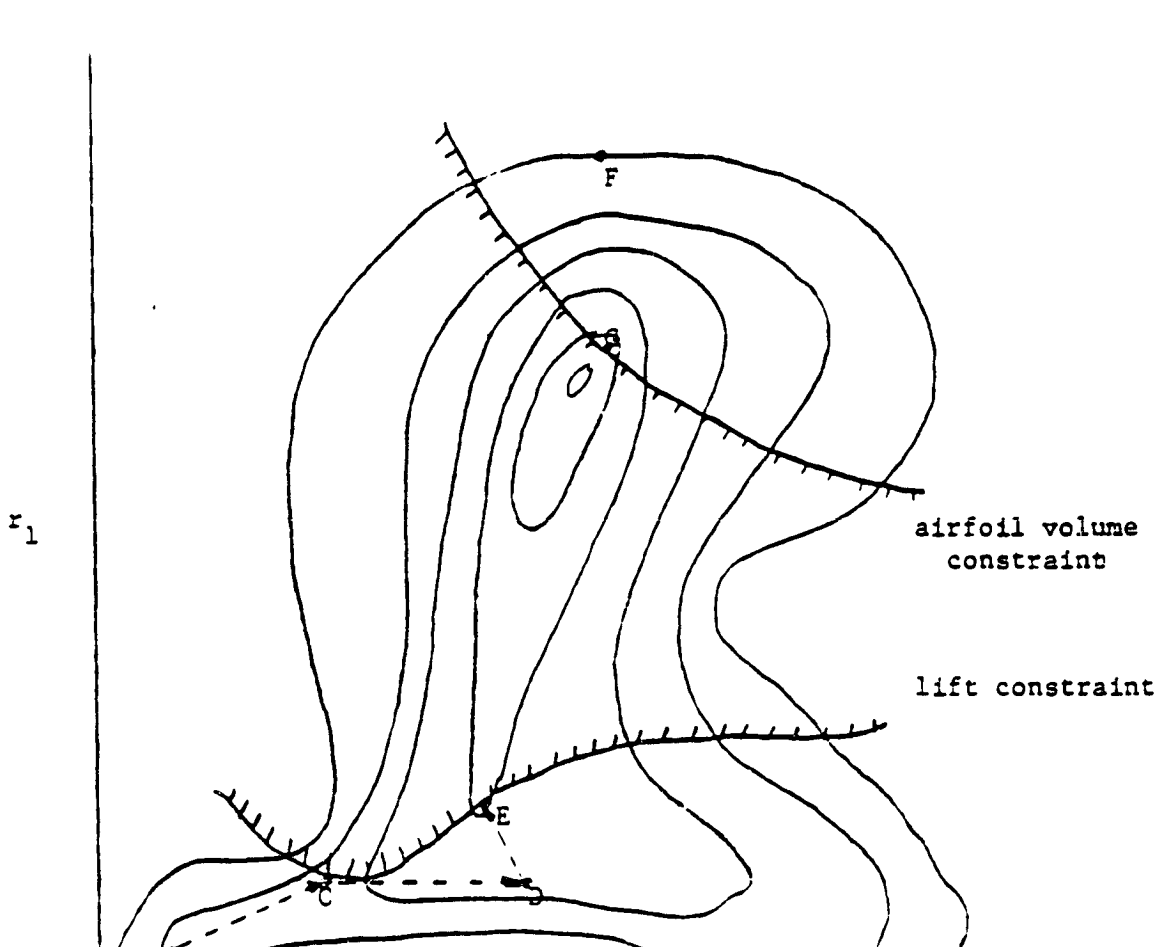
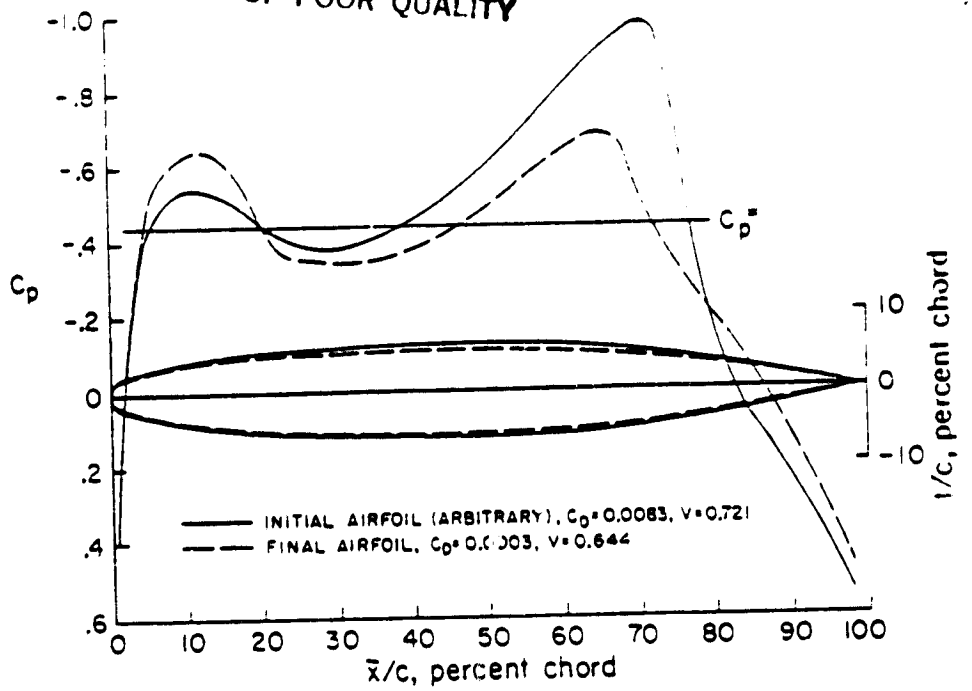
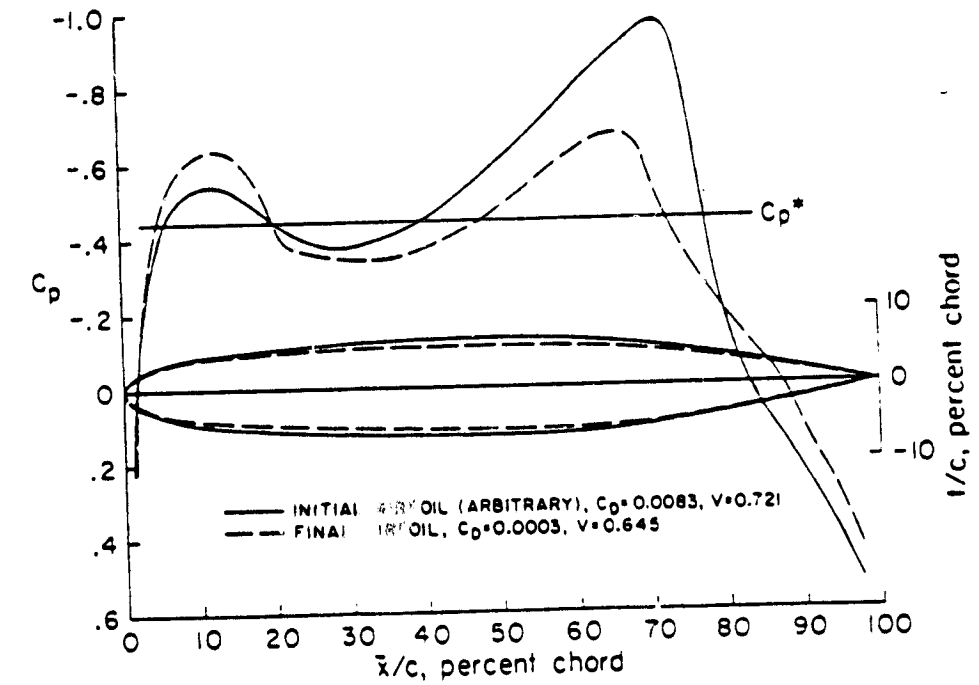


Figure 2. - Design Space Example

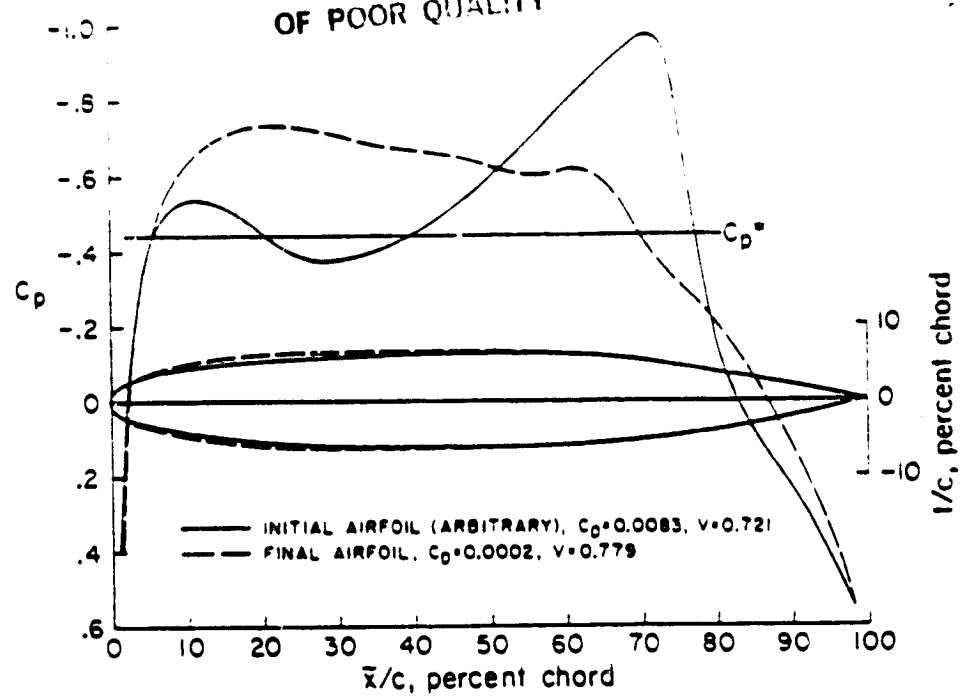


a) Constraint: Vol $\geq$ 0.4 . n = 7, M = 0.8, C_l = 0.0

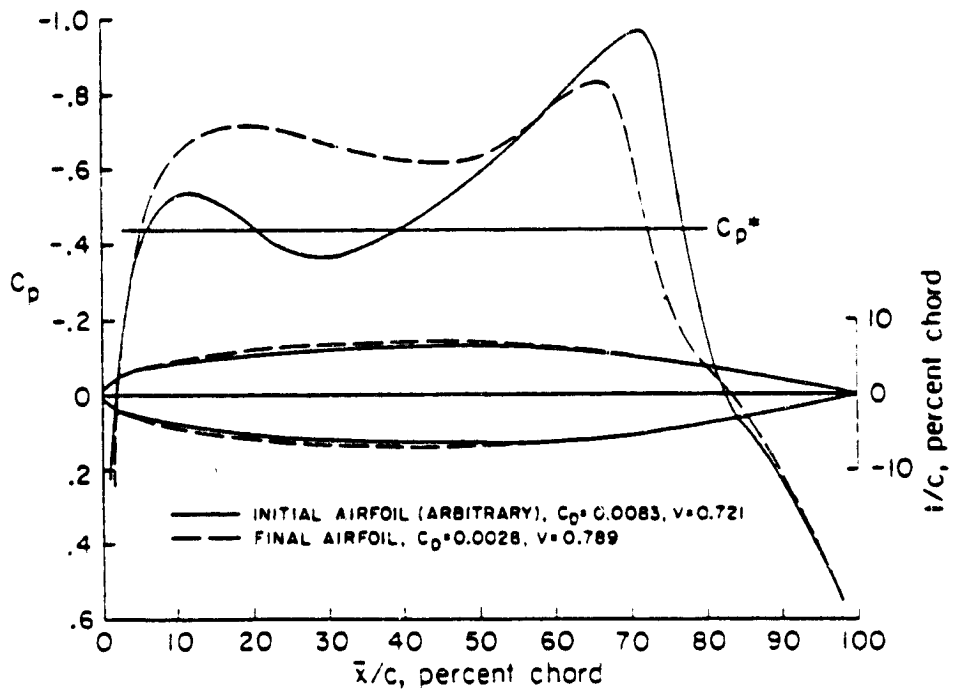


b) Constraint: Vol $\geq$ 0.6 . N = 7, M = 0.8, C_l = 0.0

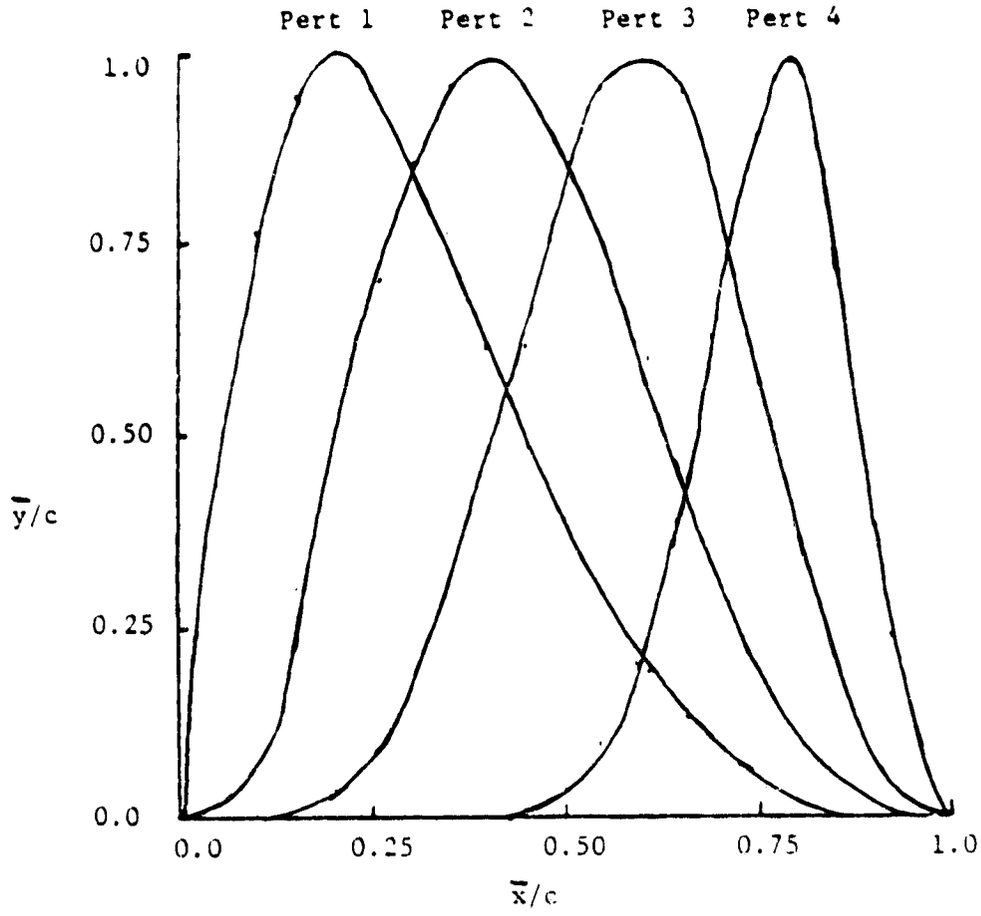
Figure 3. - Drag Minimization



c) Constraints: Vol $\geq$ 0.7, $|Ku| \leq 100$, (t/c) .125 at $(\bar{x}/c) = 0.41$
n = 7. M = 0.8. C_L = 0.0



d) Constraints: Vol $\geq$ 0.7, $|Ku| \leq 100$, (t/c) .125 at $(\bar{x}/c) = 0.51$
n = 7, M =0.8, C_l = 0.0



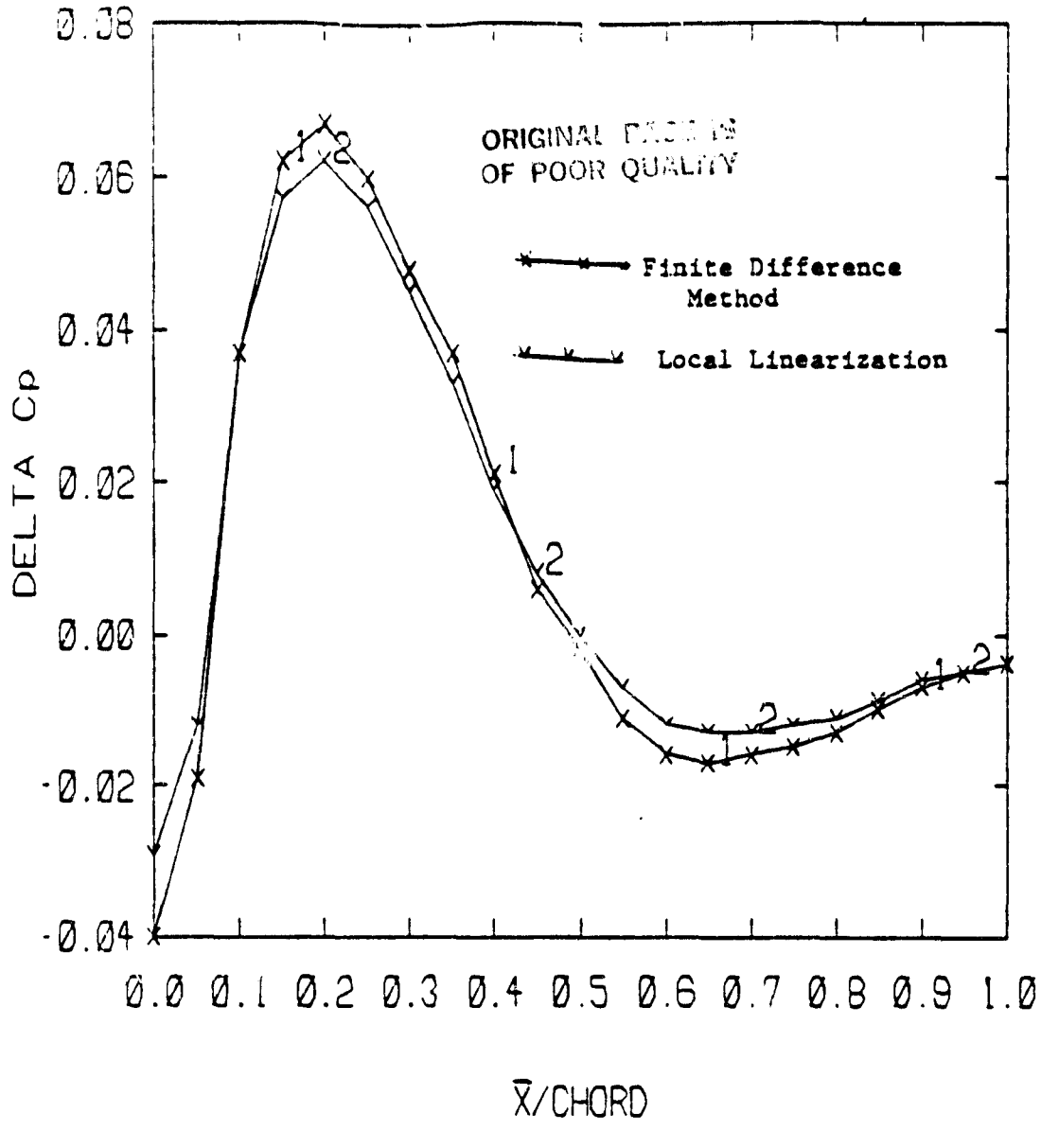
$$\text{Pert 1} = \left(\sin\left(\pi\left(\bar{x}/c\right)^{.4M}\right)\right)^3$$

$$\text{Pert 2} = \left(\sin\left(\pi\left(\bar{x}/c\right)^{.357}\right)\right)^3$$

$$\text{Pert 3} = \left(\sin\left(\pi\left(\bar{x}/c\right)^{.357}\right)\right)^3$$

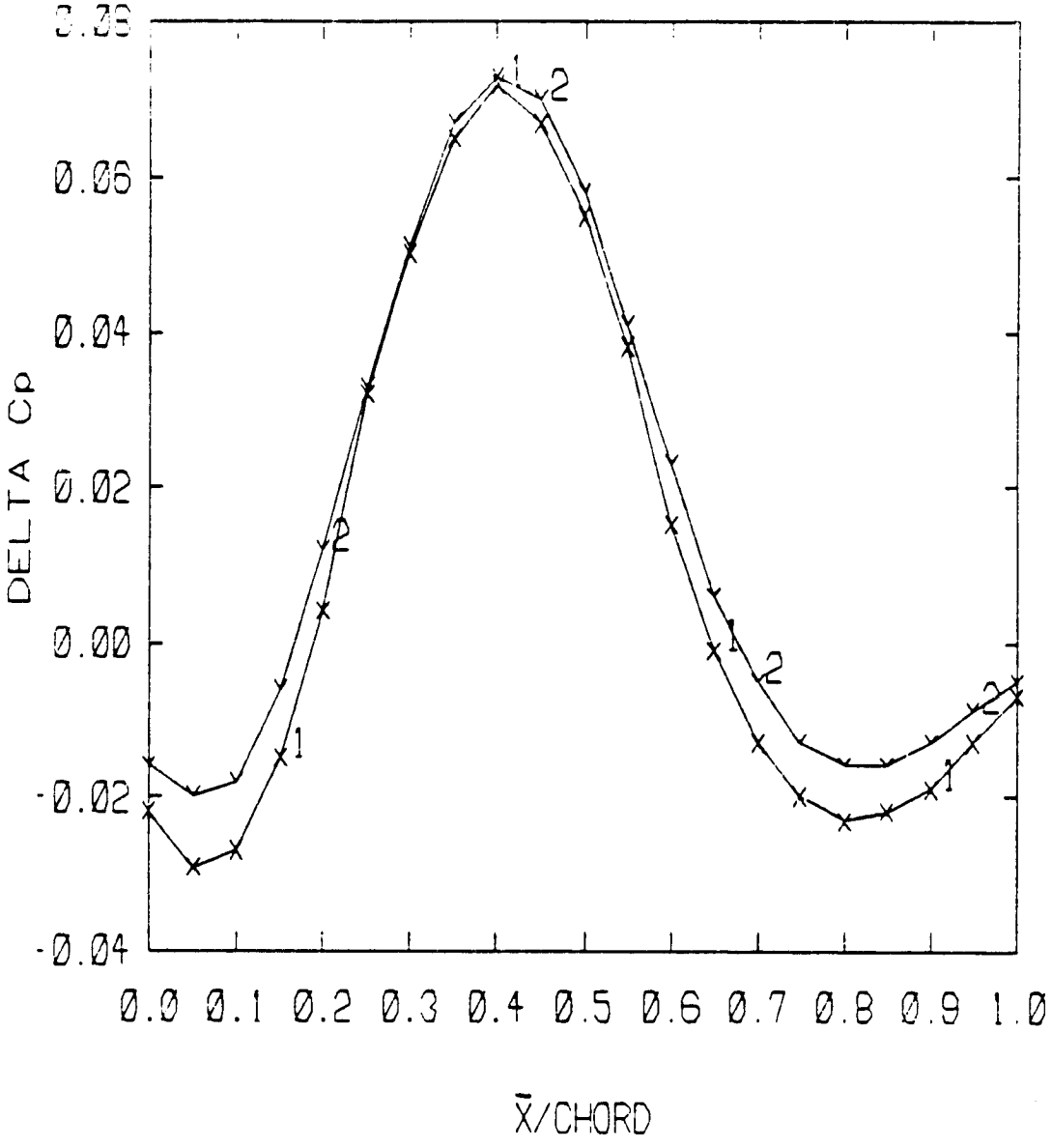
$$\text{Pert 4} = \left(\sin\left(\pi\left(\bar{x}/c\right)^{1.386}\right)\right)^3$$

Figure 4. - Shape Functions

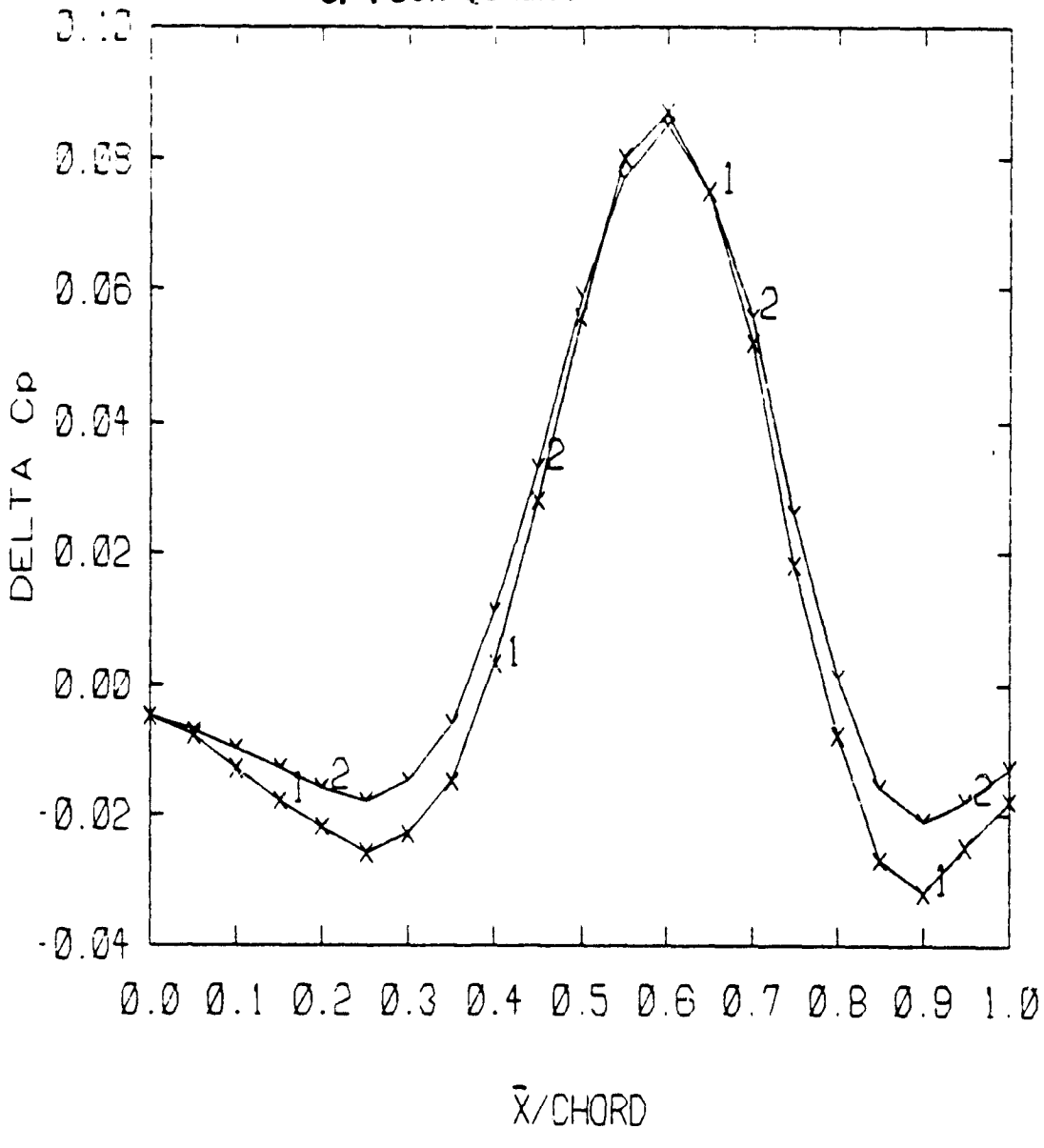


a) Perturbation 1

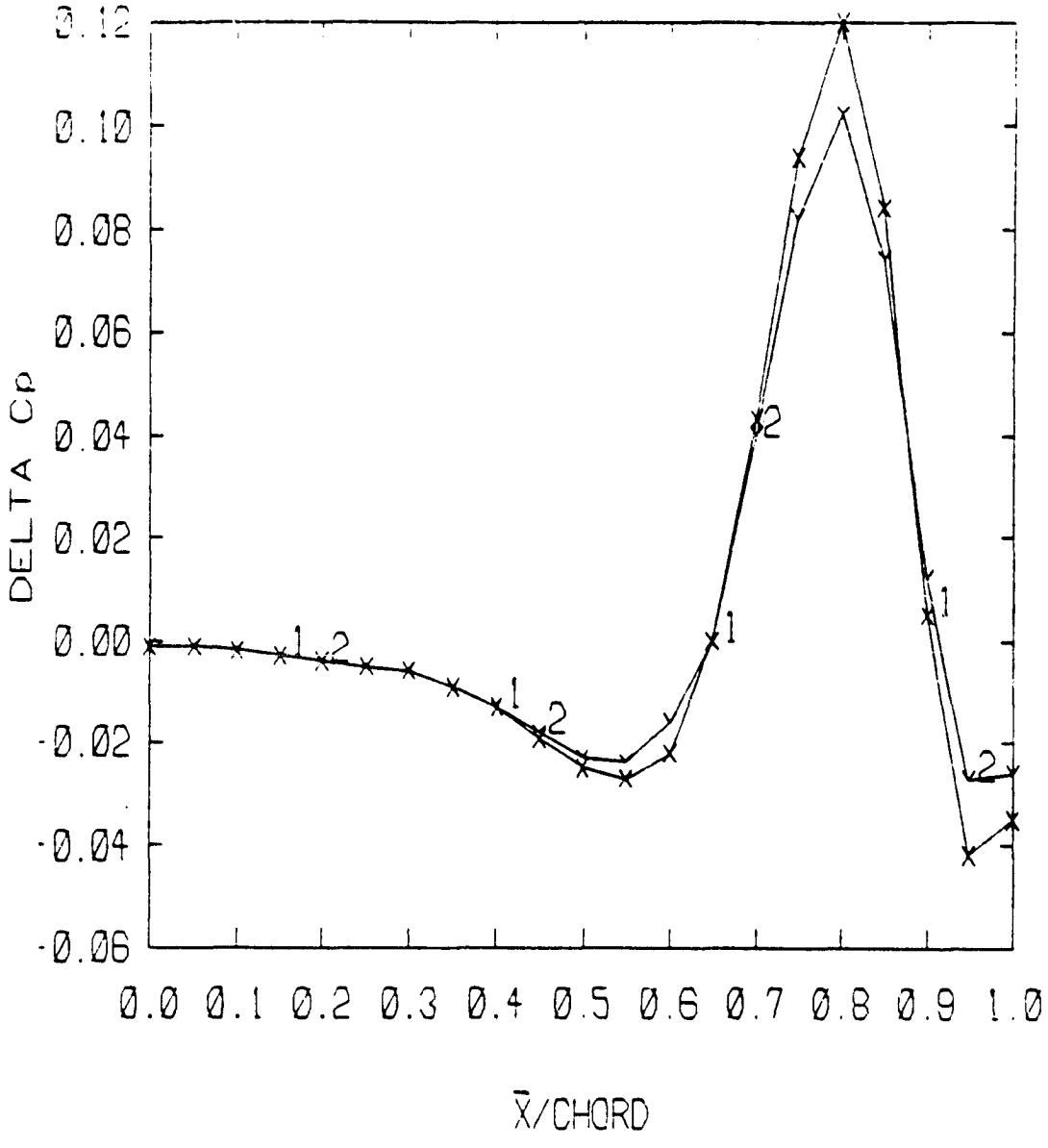
Figure 5. - Perturbation in Cp as Computed Using Local Linearization and Finite Difference Method. M = 0.75, (t/c) = 0.08



b) Perturbation 2



c) Perturbation 3



d) Perturbation 4

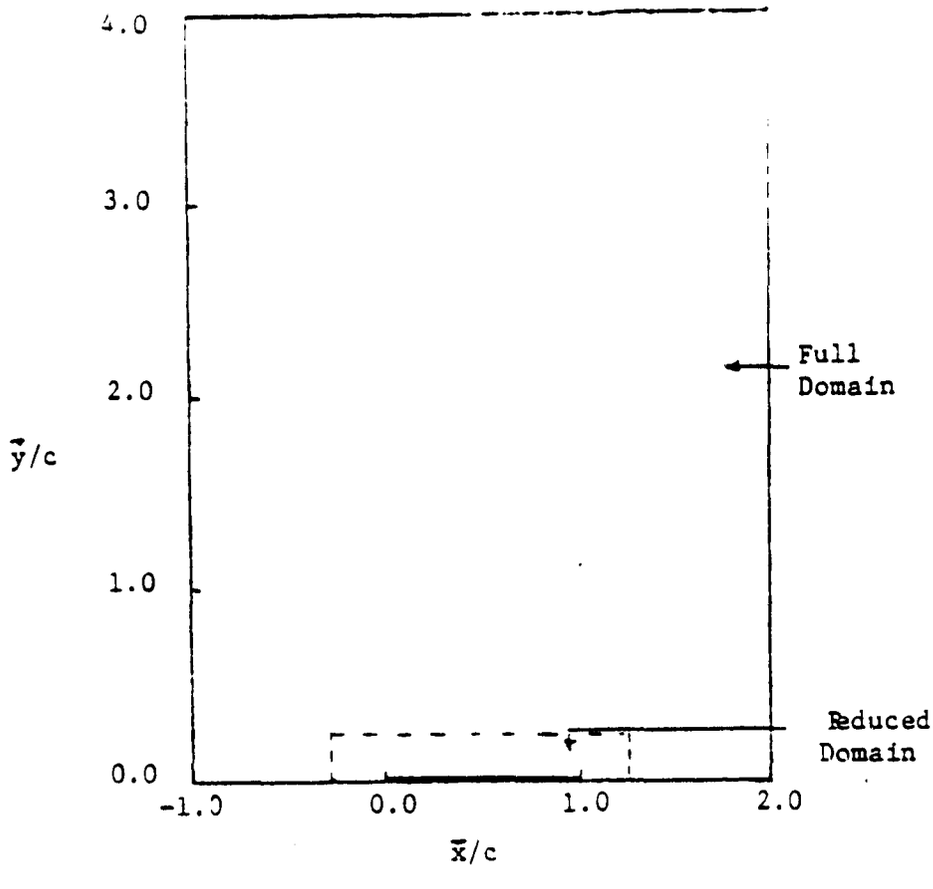


Figure 6. - Comparison of Computational Domains

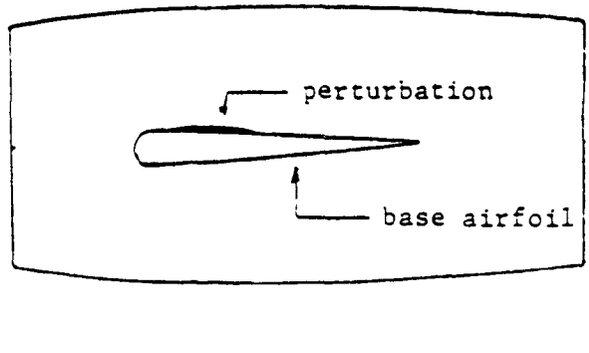
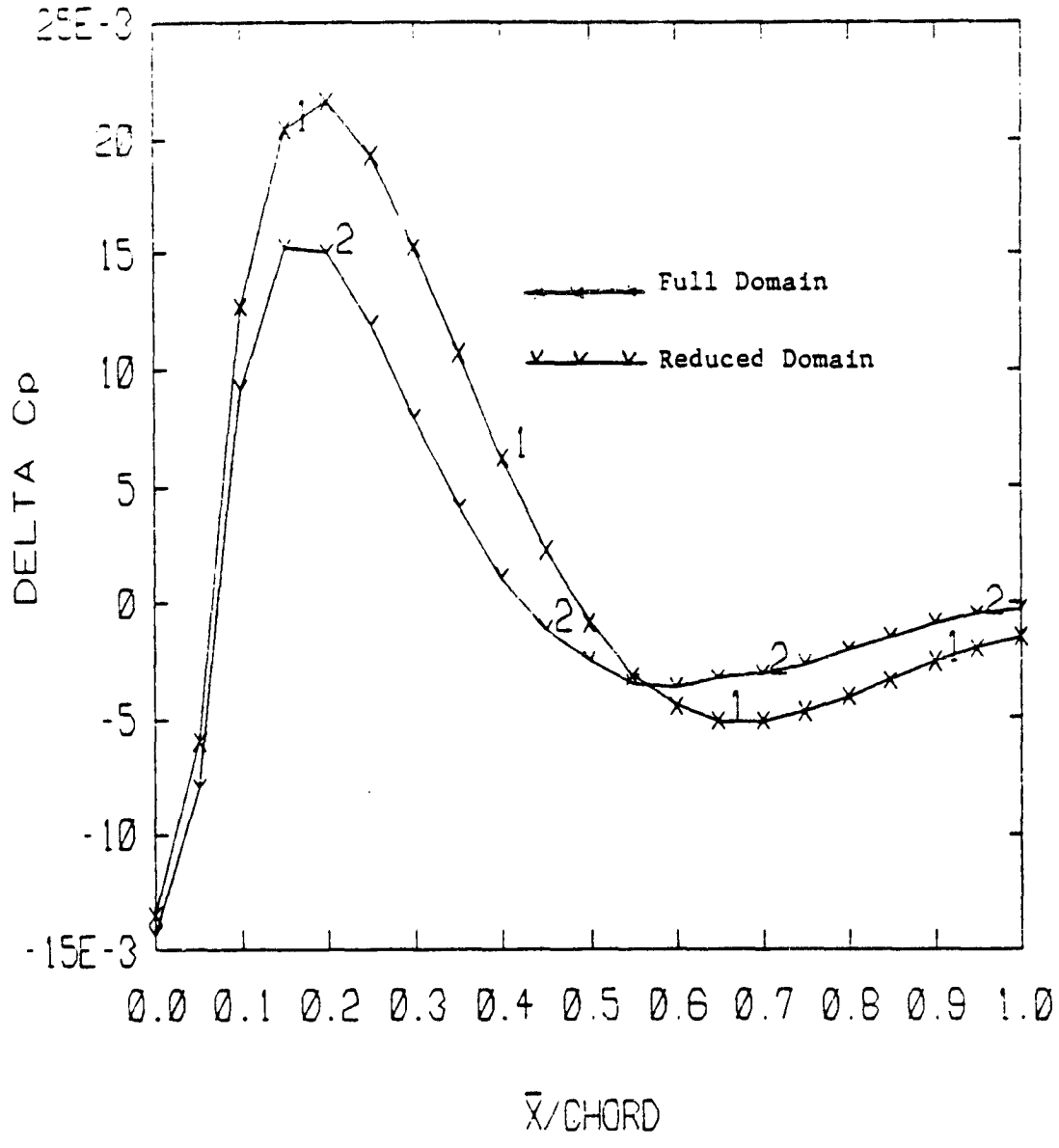
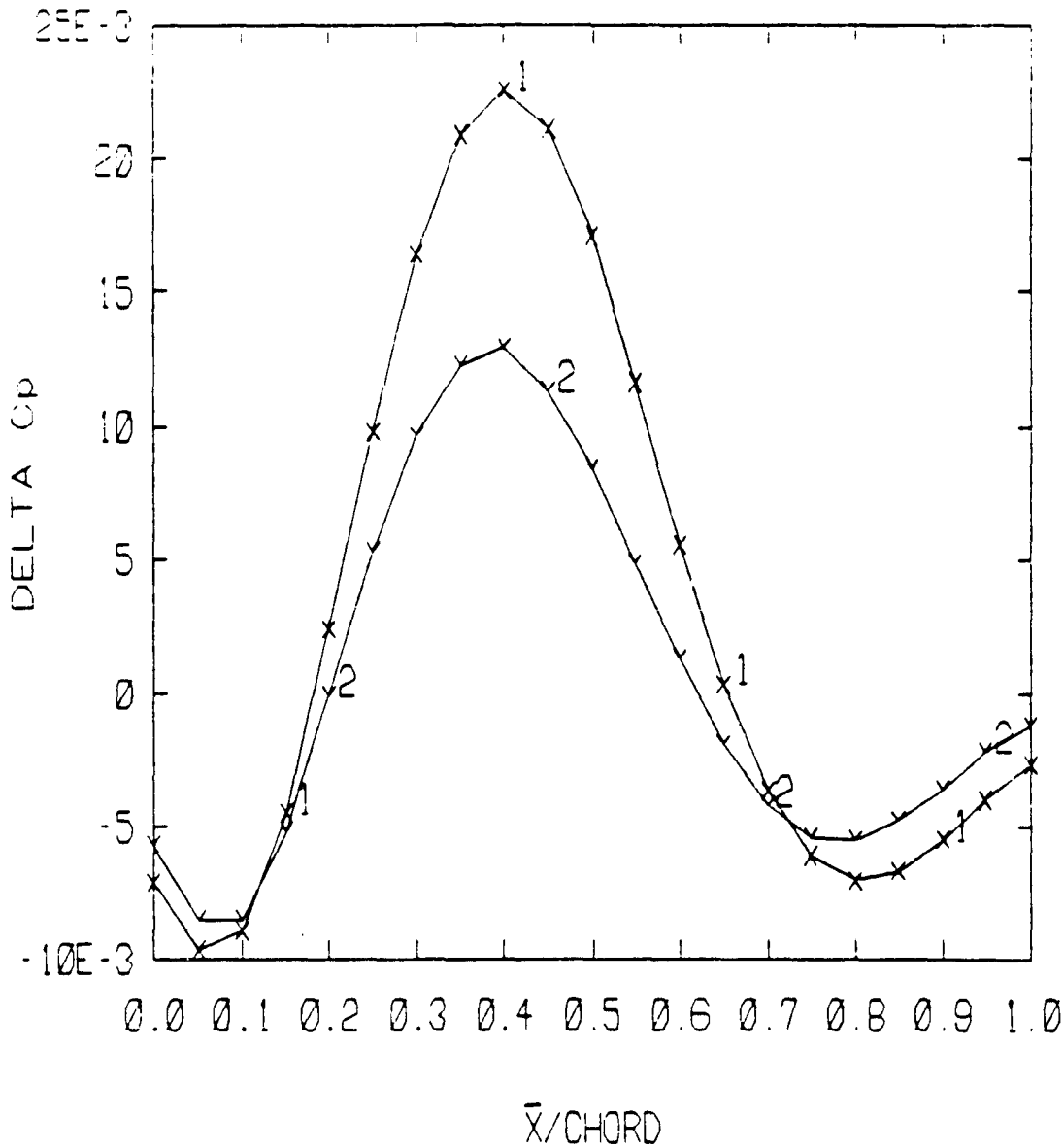


Figure 7. - Airfoil in Hypothetical Windtunnel

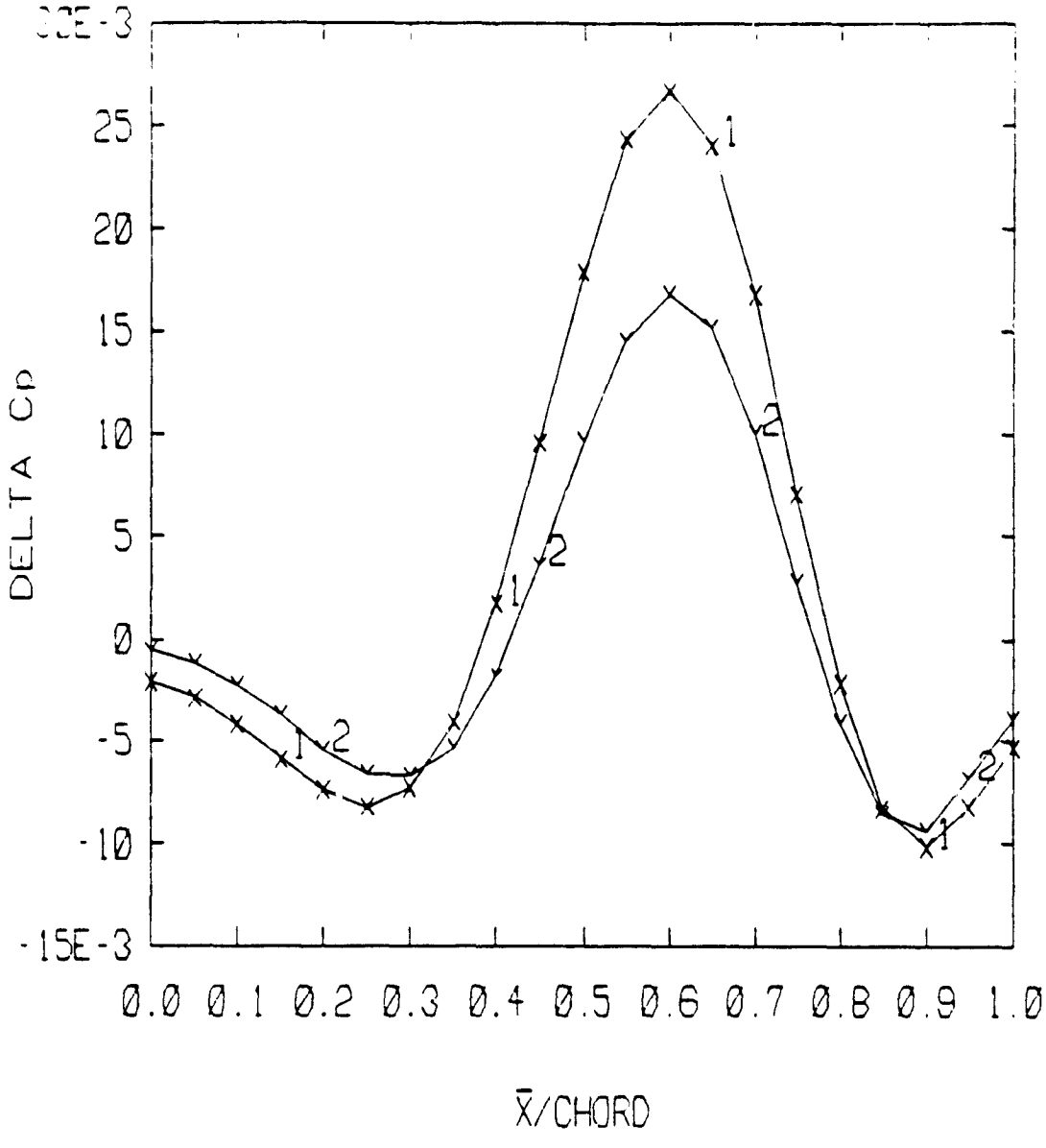


a) Pertubation 1

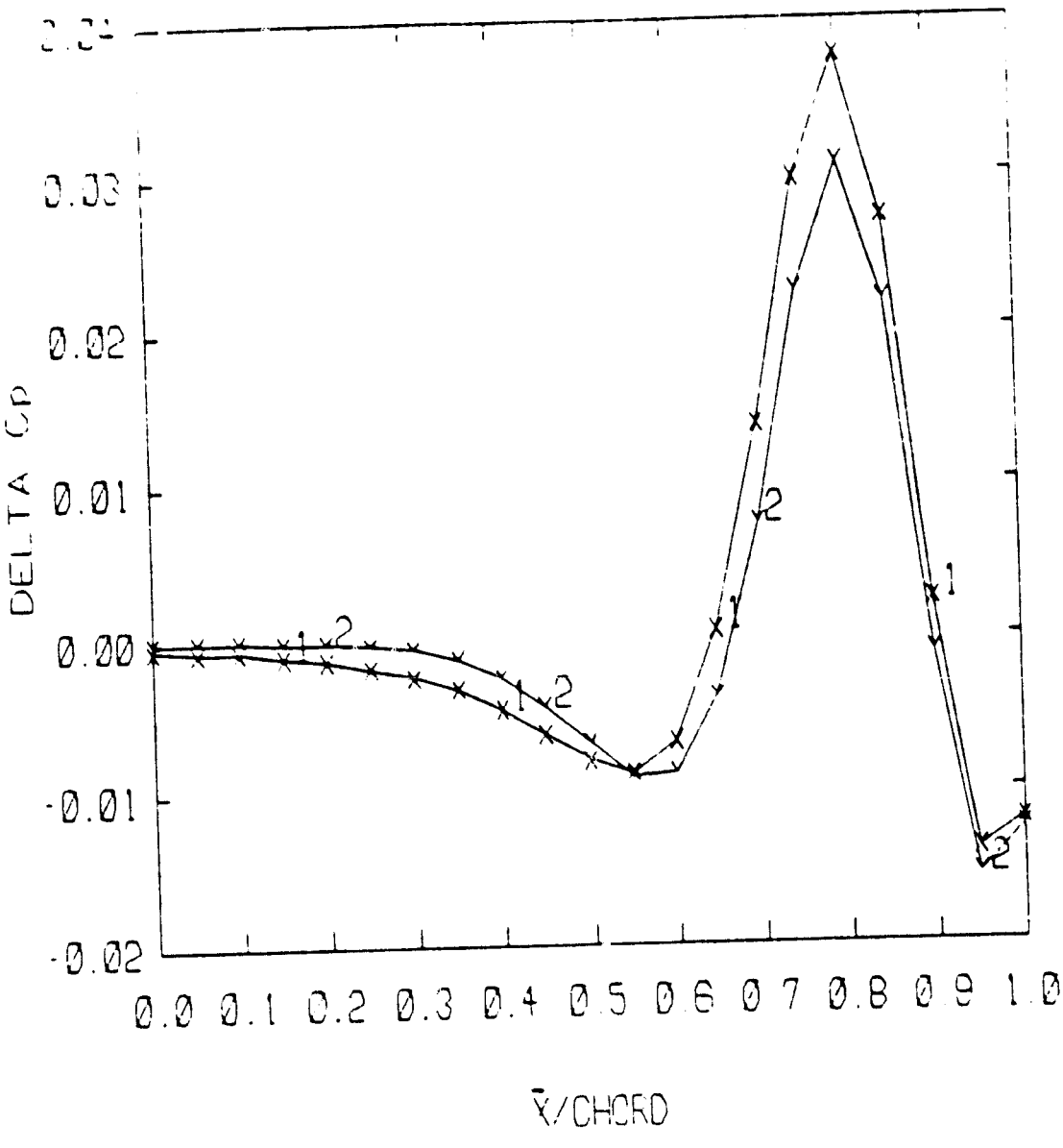
Figure 8. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Base Potential Specified at Edge of Reduced Domain. M = 0.7, (t/c) = 0.1



b) Perturbation 2



c) Perturbation 3



d) Perturbation 4

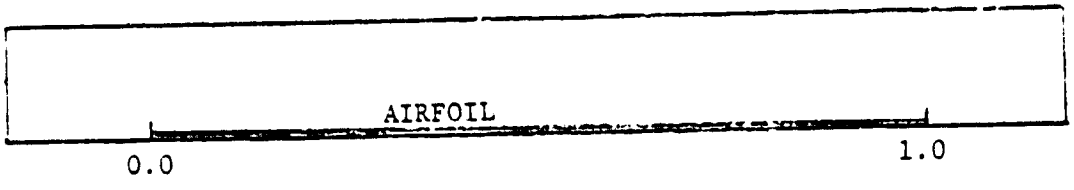
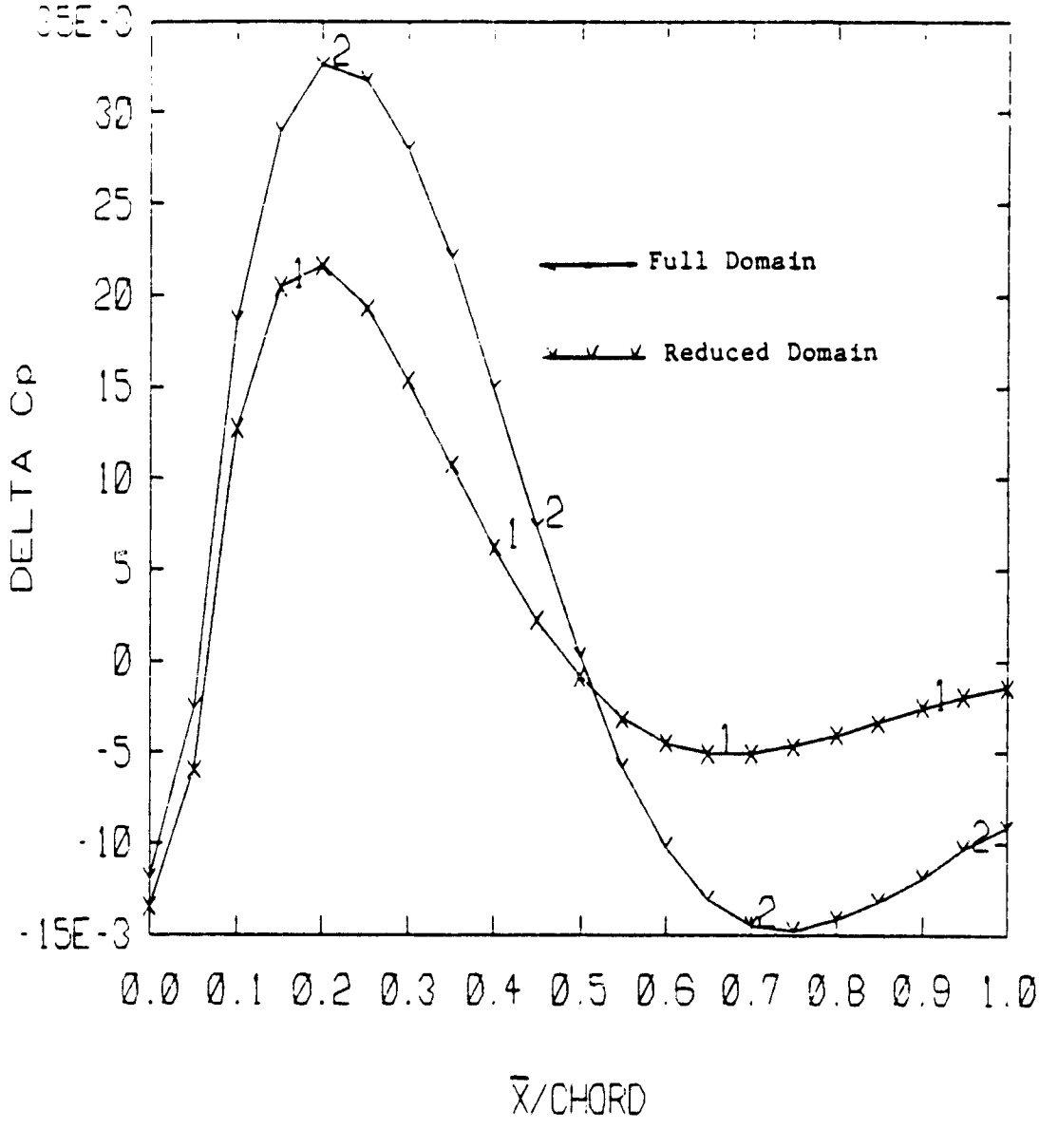
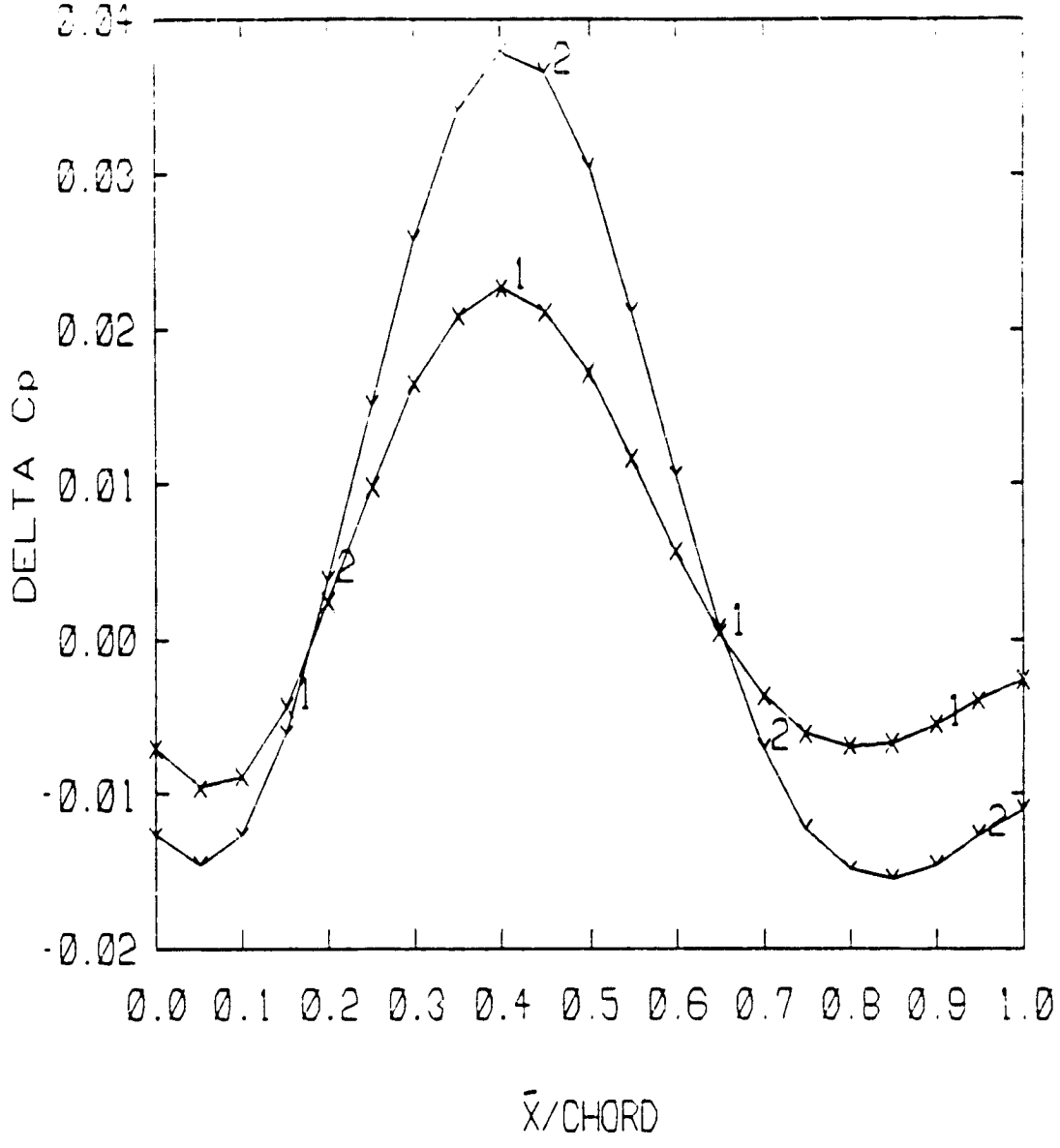


Figure 9. - Reduced Domain

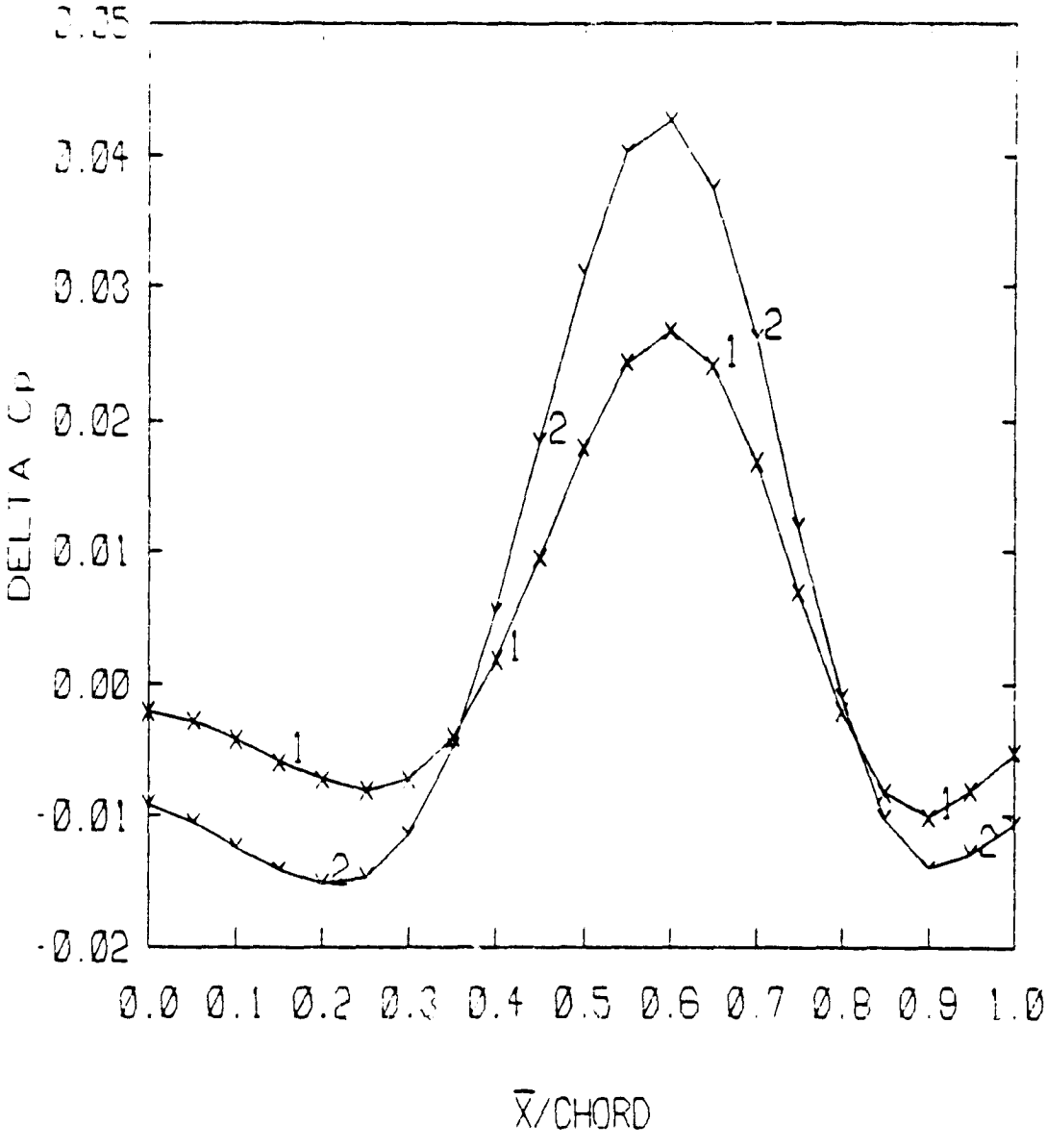


a) Perturbation 1

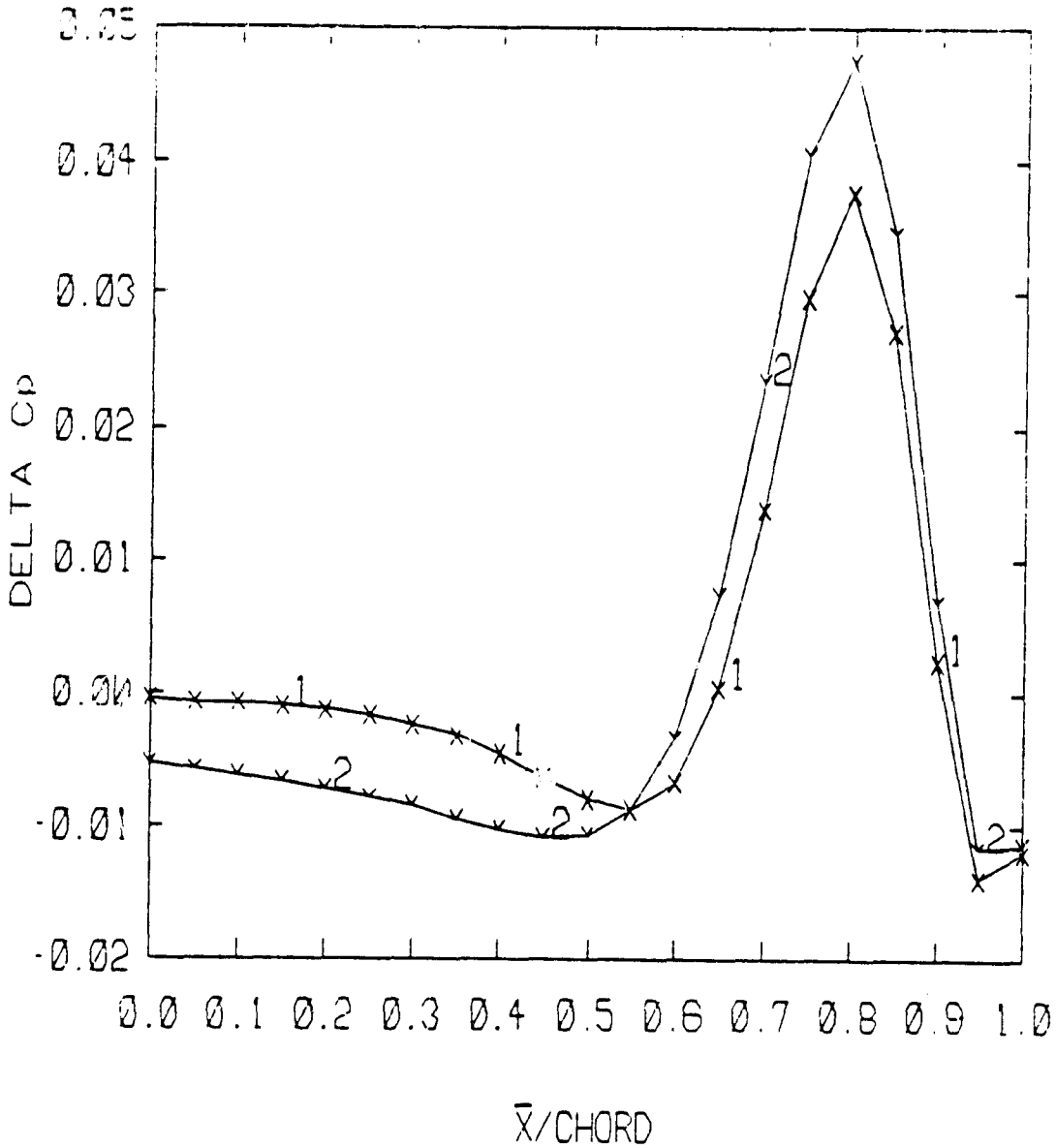
Figure 10. - Pertubation in Cp as Computed on Full Domain and Reduced Domain. Base Transverse Velocity Specified at Top of Reduced Domain. M = 0.7, (t/c) = 0.1



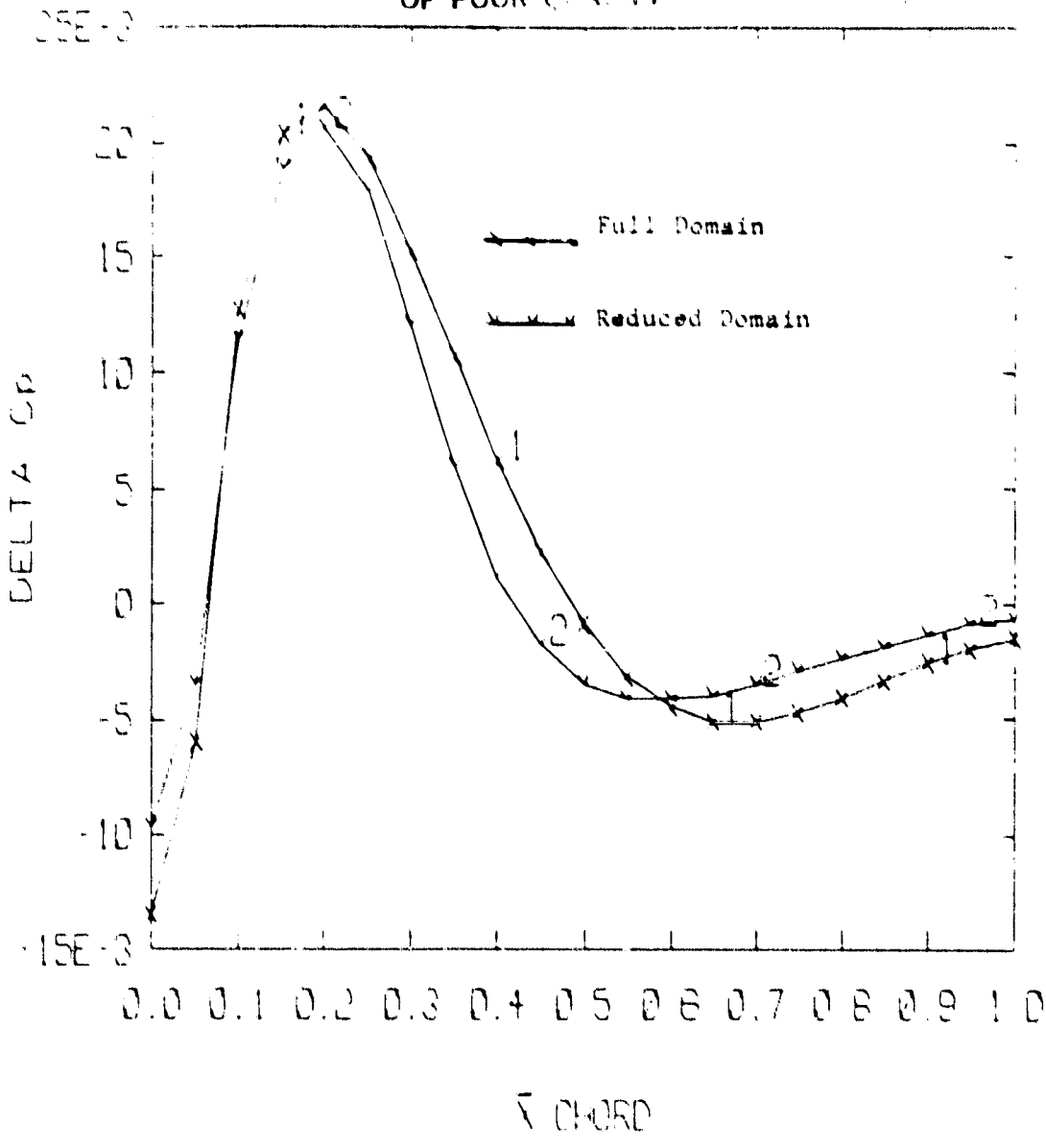
b) Perturbation 2



c) Perturbation 3



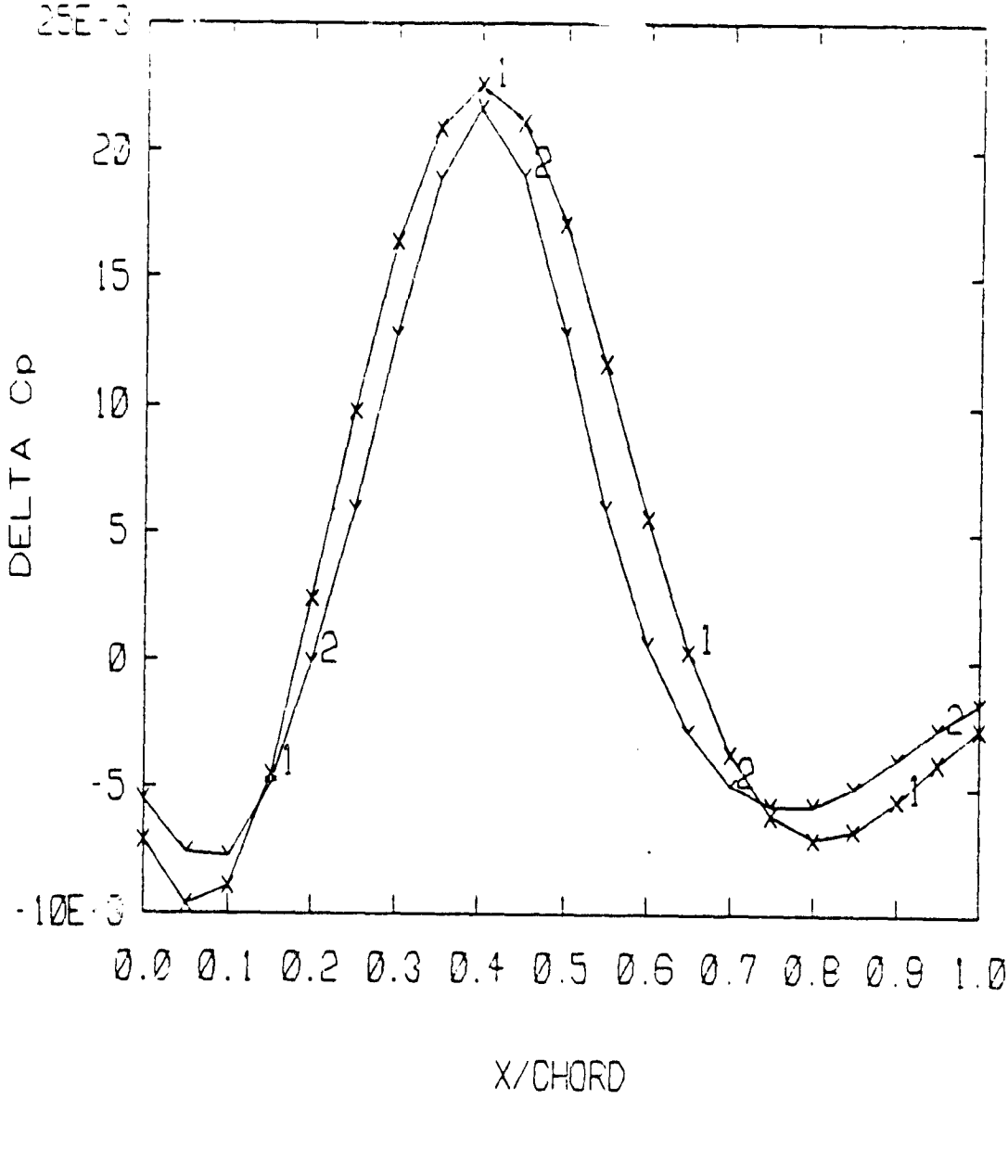
d) Perturbation 4



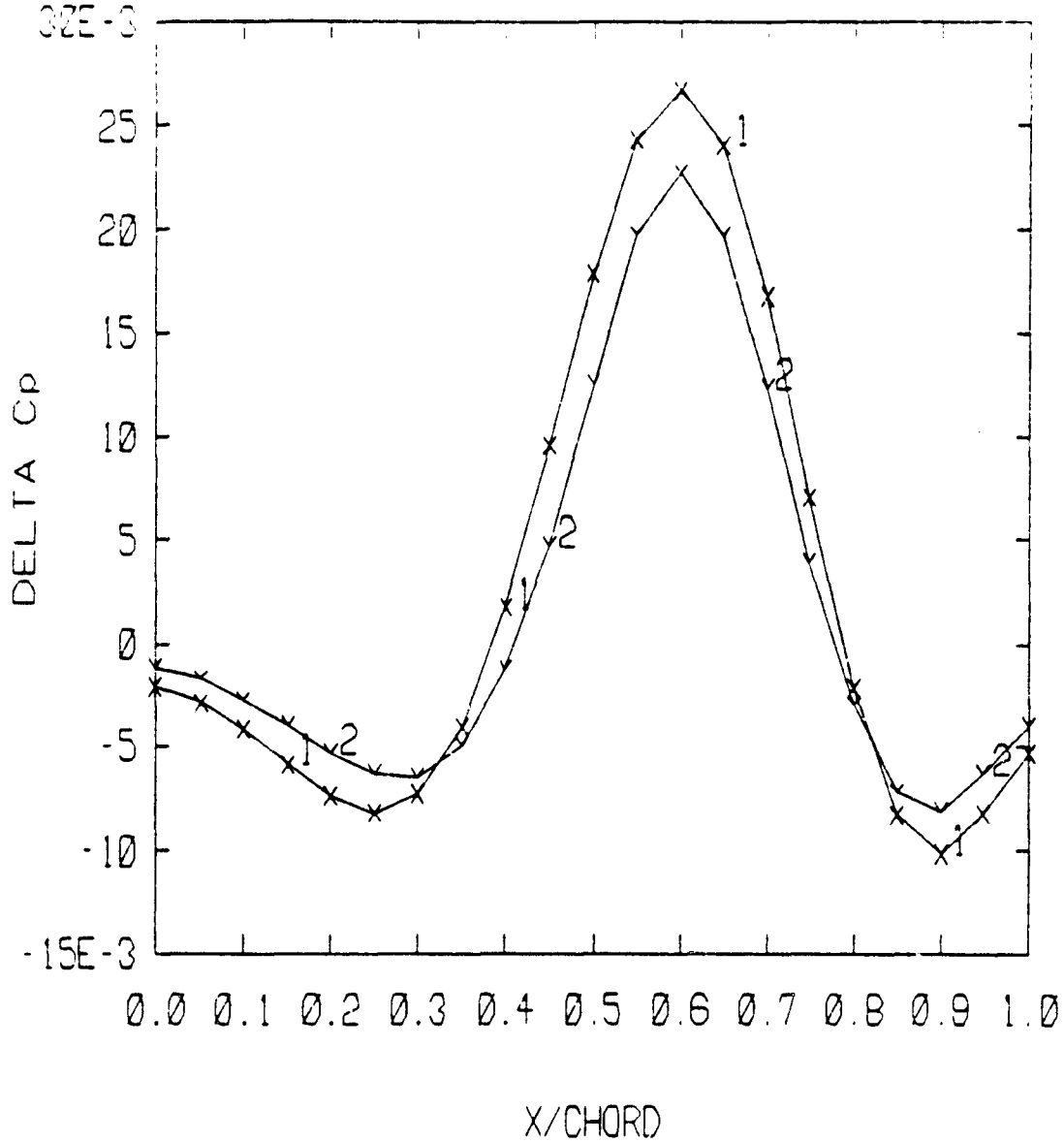
a) Perturbation 1

Figure 11. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Perturbation Potential Added to Base Potential at Edge of Reduced Domain. M = 0.7, (t/c) = 0.1

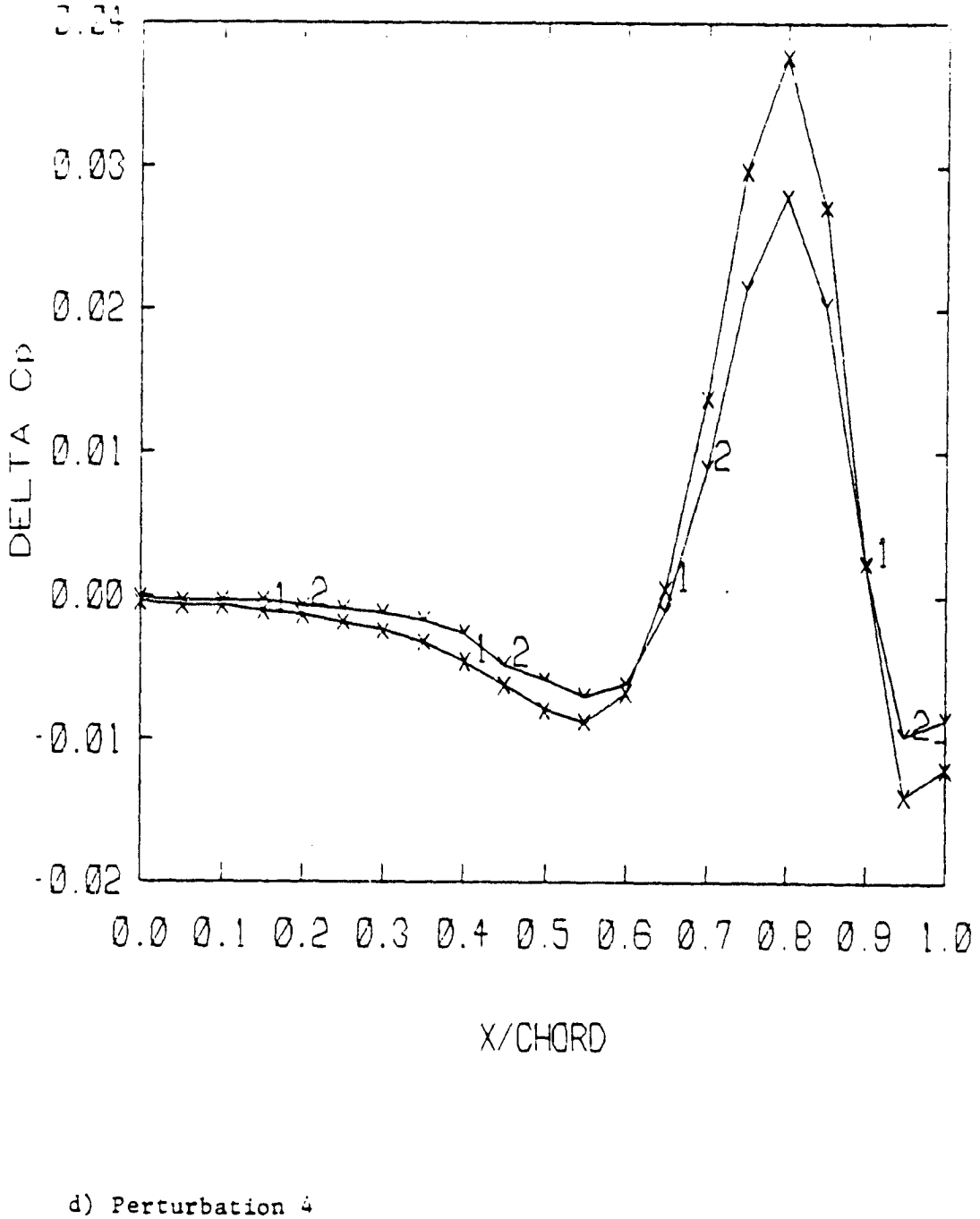
ORIGINAL PAGE IS
OF POOR QUALITY



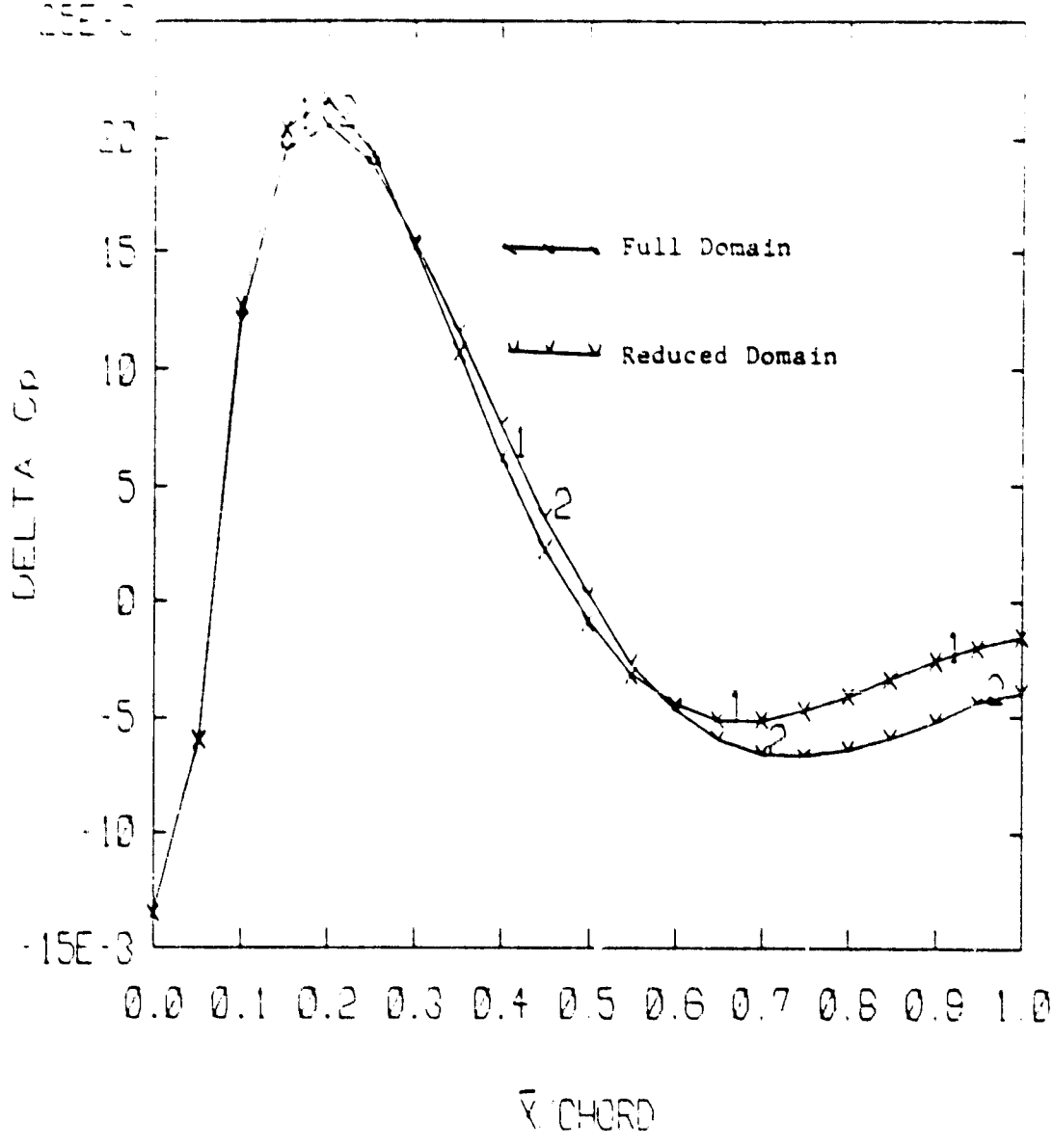
b) Perturbation 2



c) Perturbation 3

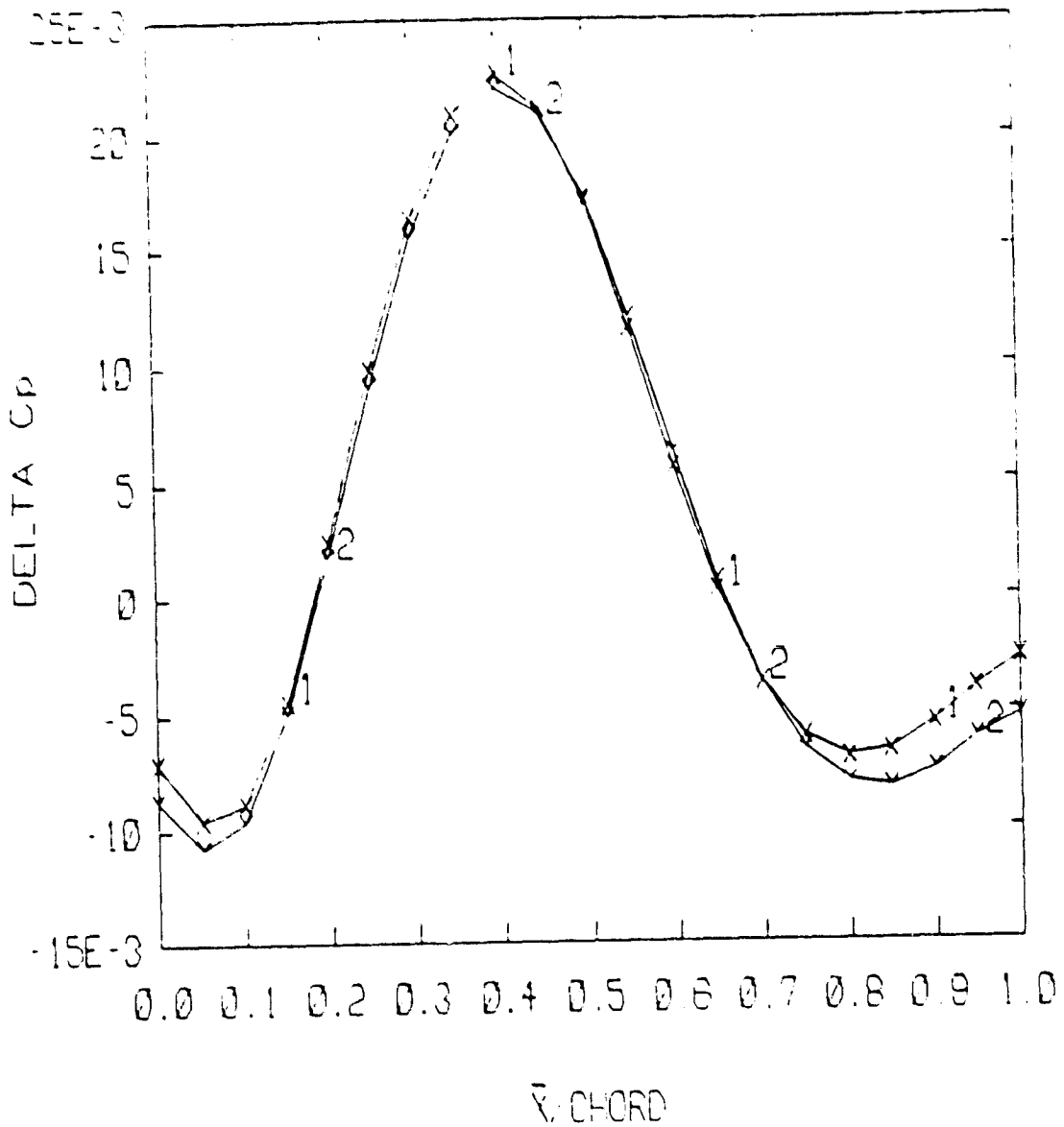


d) Perturbation 4

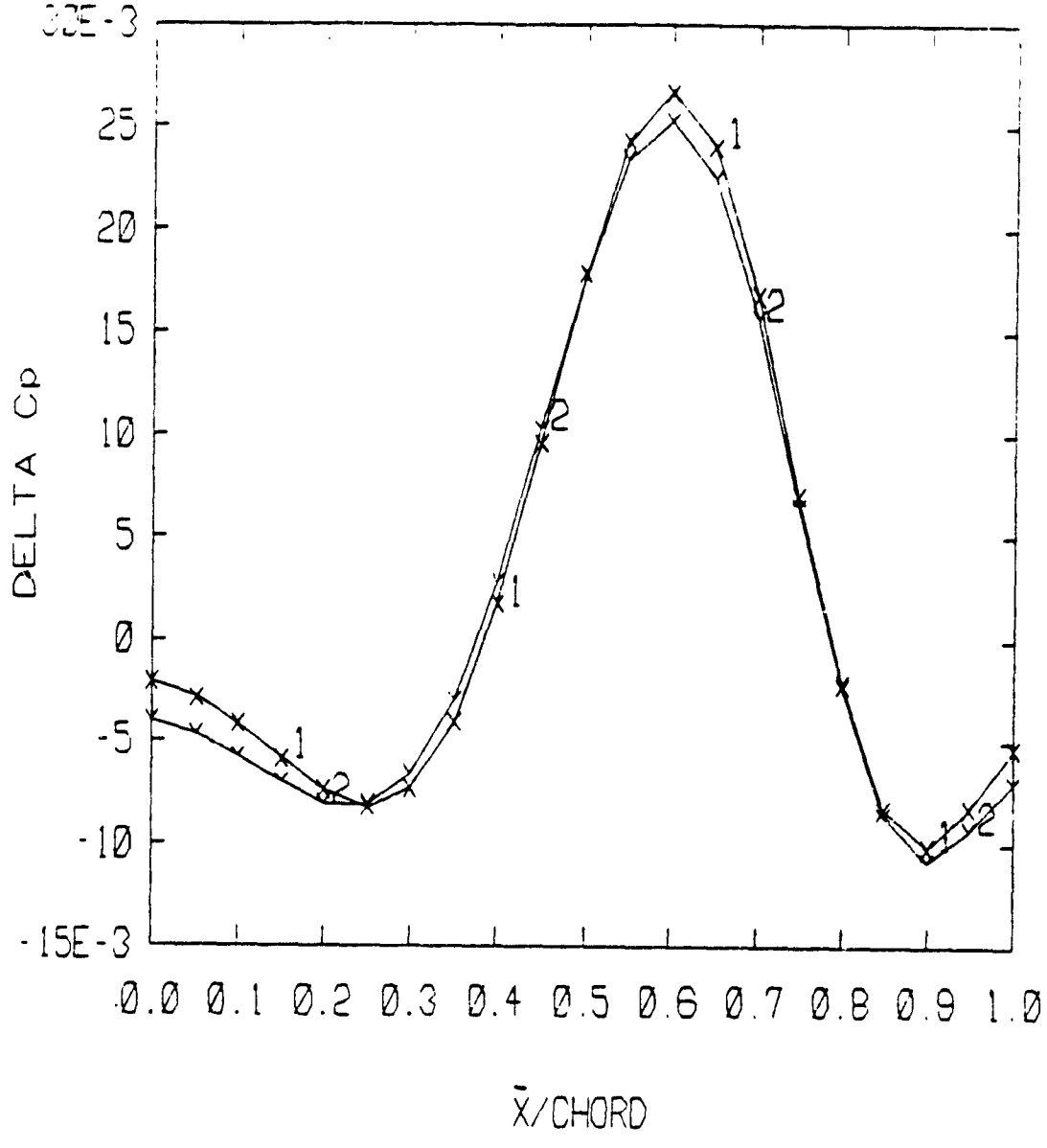


a) Perturbation 1

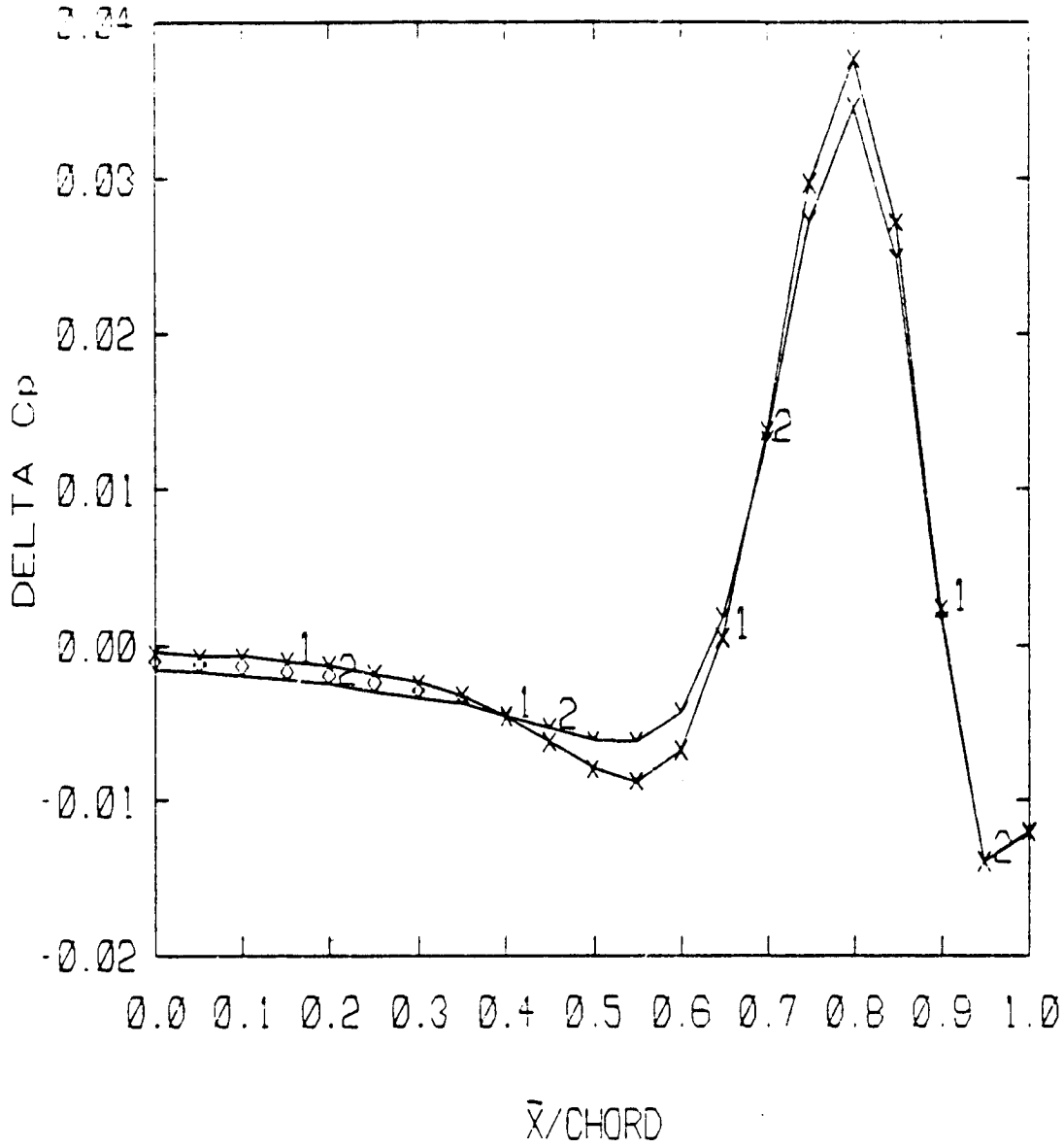
Figure 12. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Perturbation Transverse Velocity Added to Base Transverse Velocity at Top of Reduced Domain. M = 0.7, (t/c) = 0.1



b) Perturbation 2



c) Perturbation 3



d) Perturbation 4

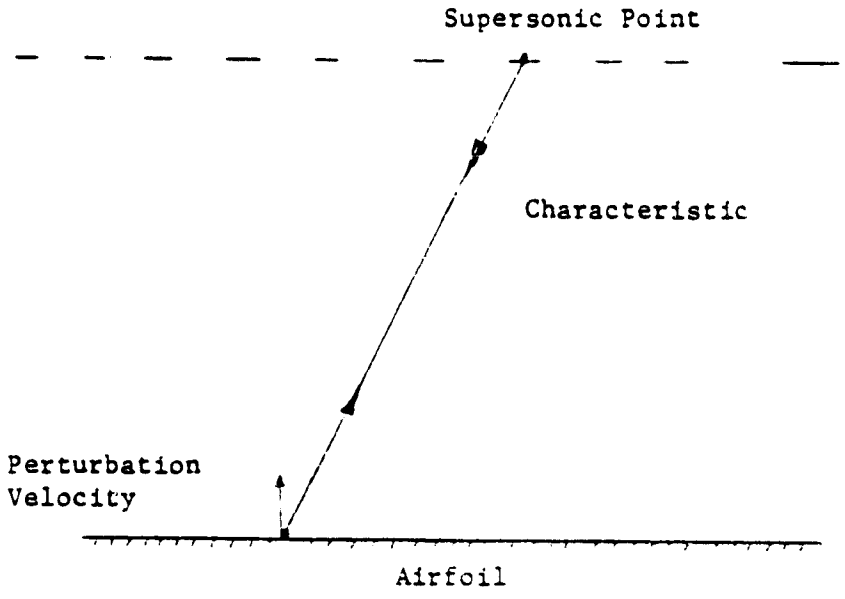
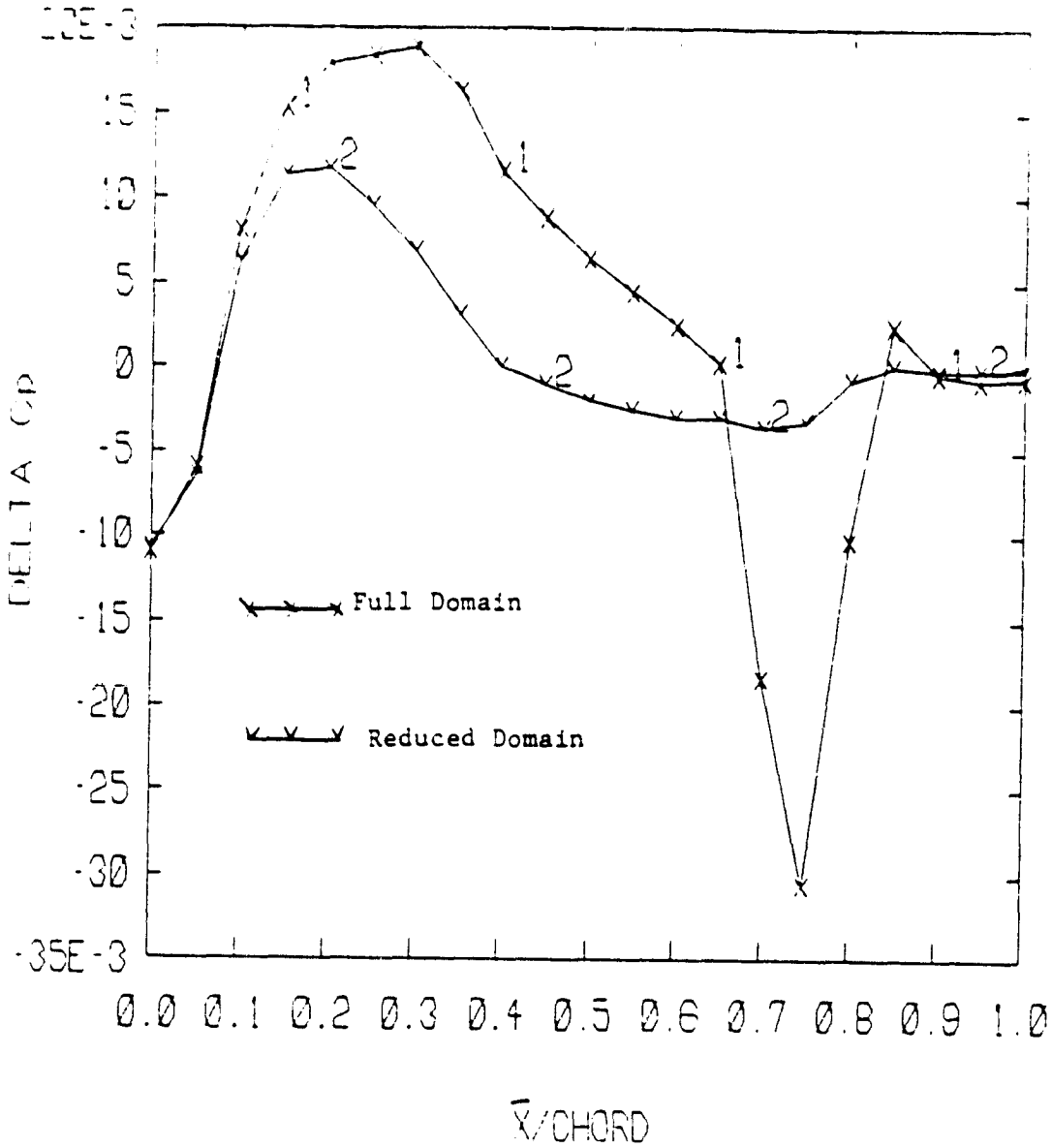
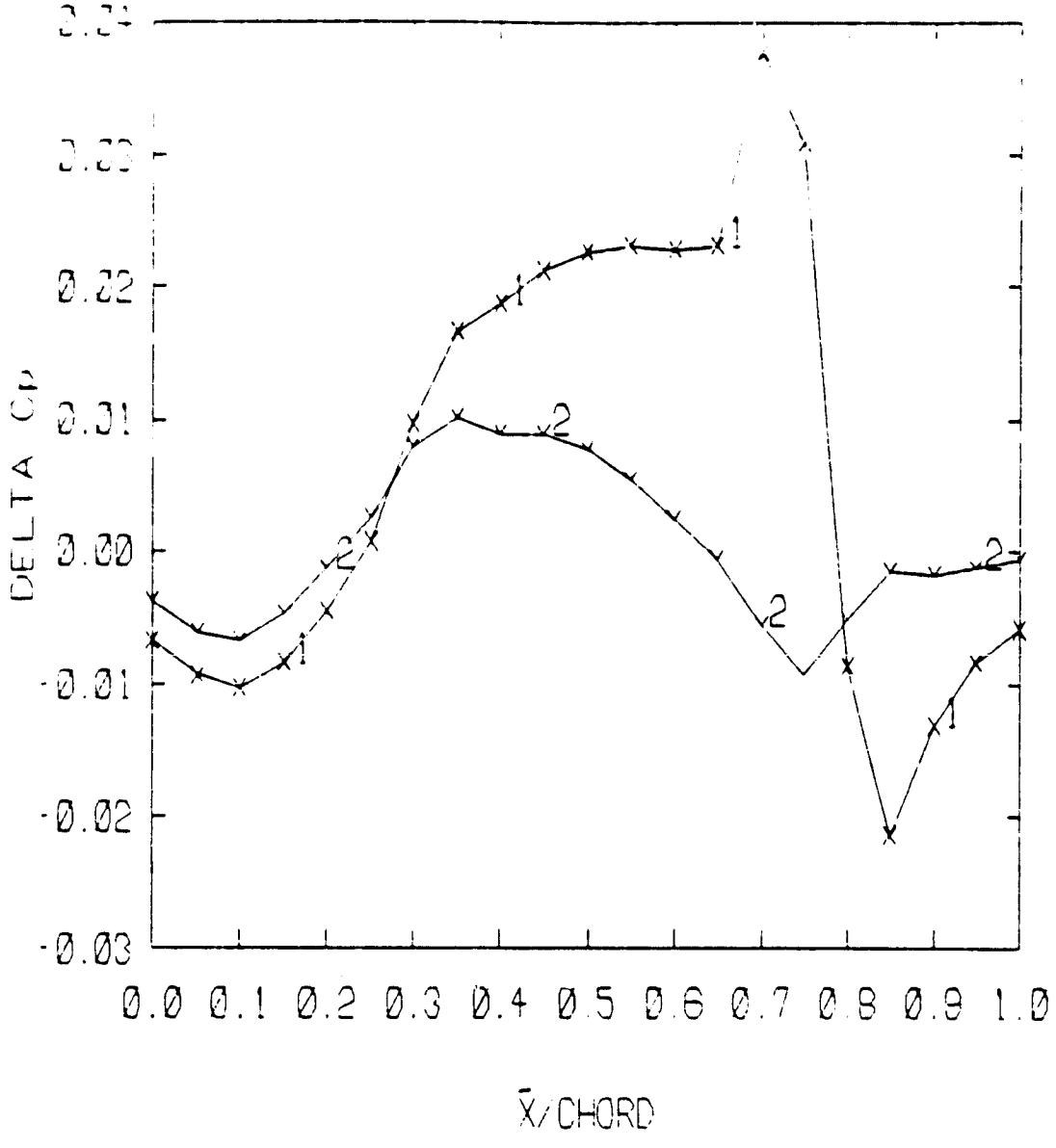


Figure 13. - Use of Characteristic Boundary Conditions in Supersonic Flow

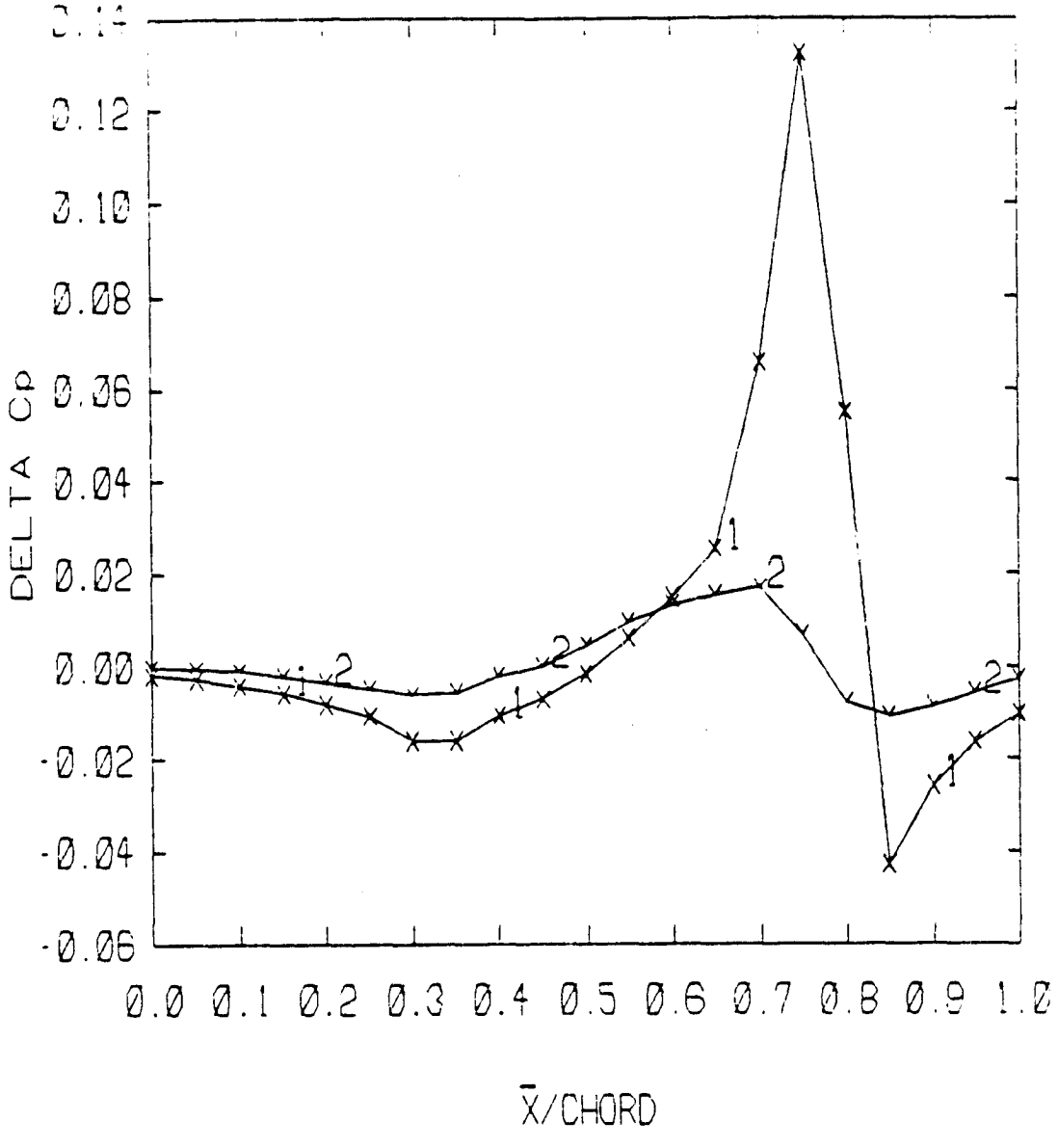


a) Perturbation 1

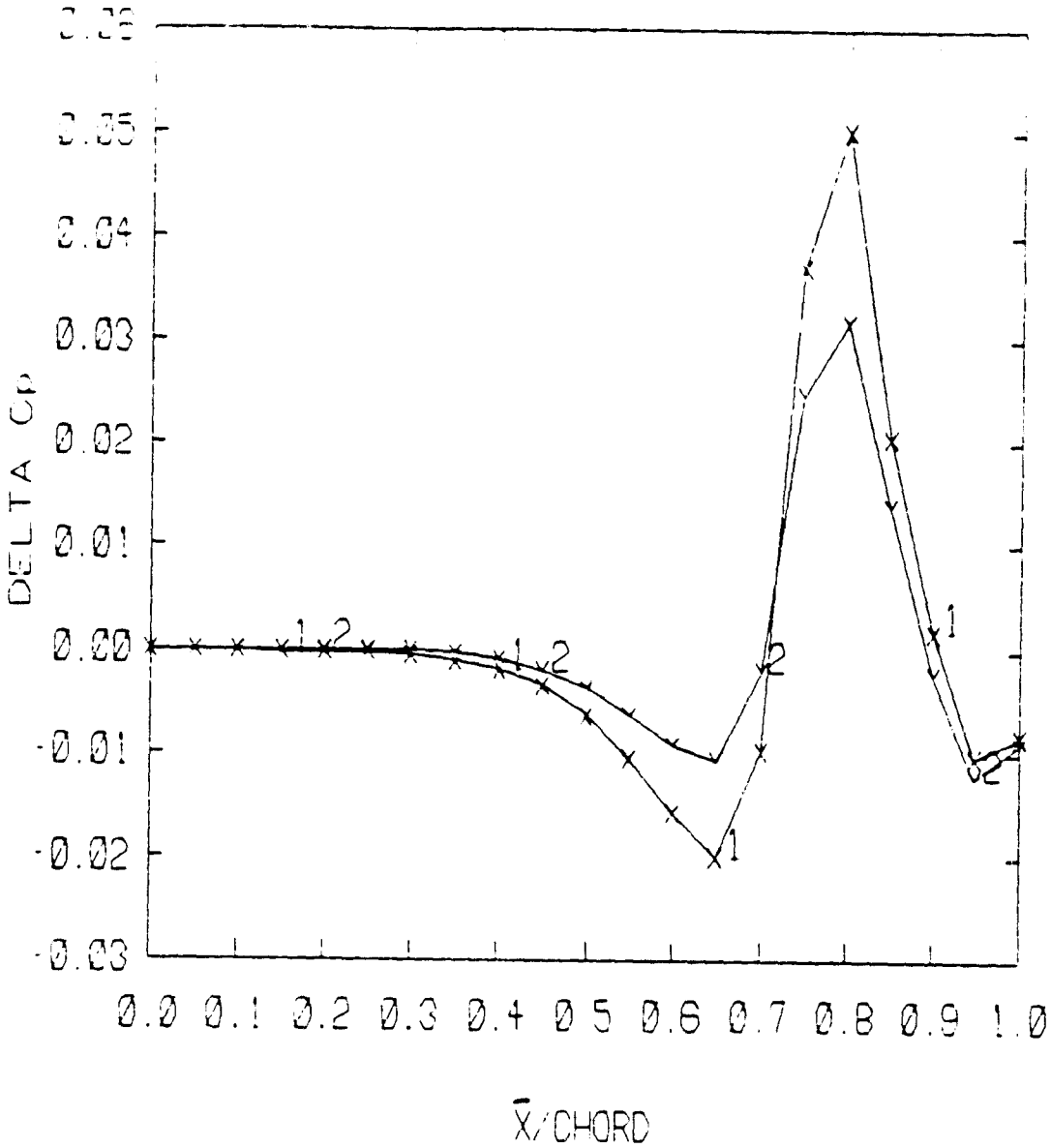
Figure 14. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Base Potential Specified at Edge of Reduced Domain. M_∞ =0.82, (t/c) = 0.1



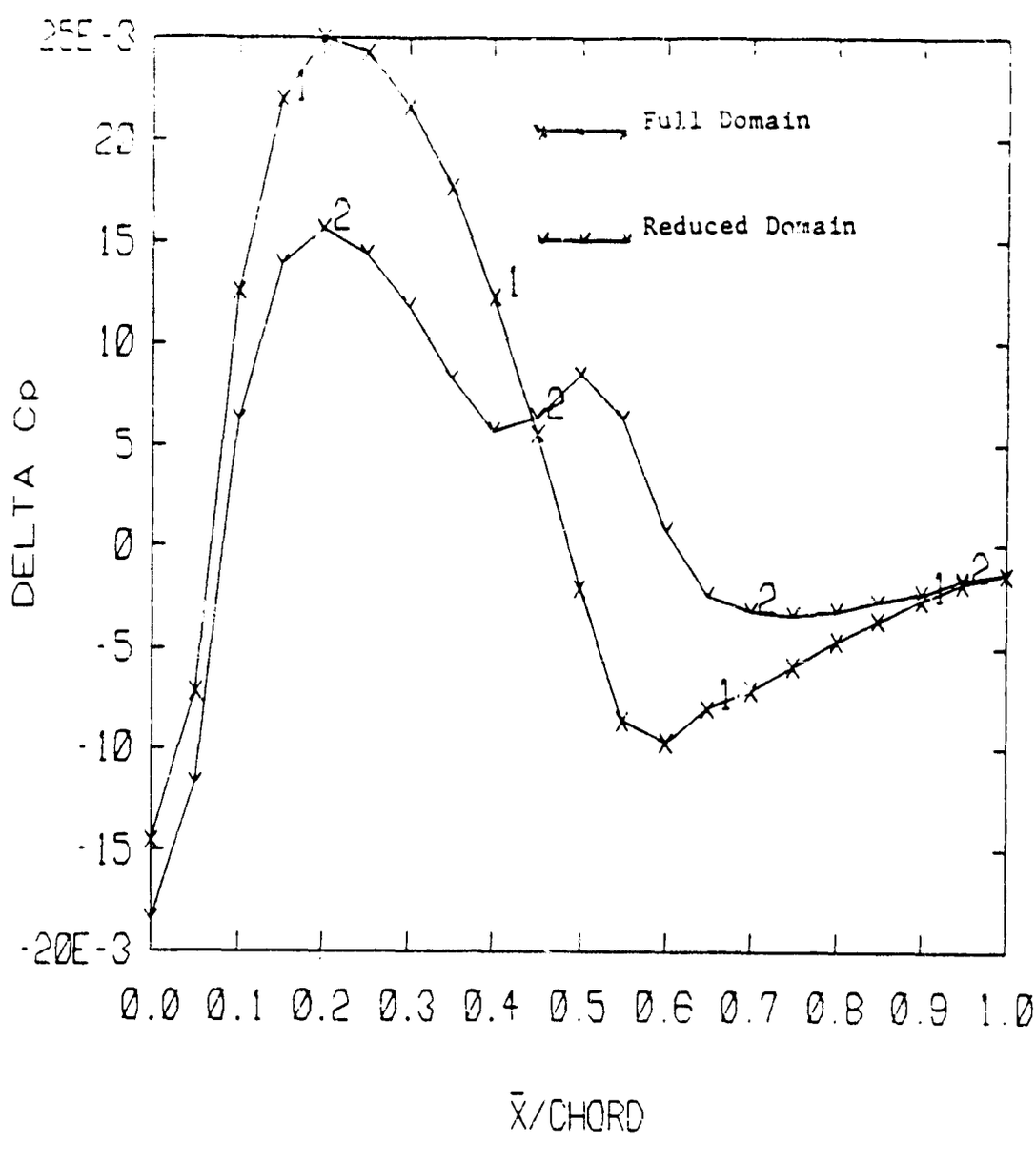
b) Perturbation 2



c) Perturbation 3

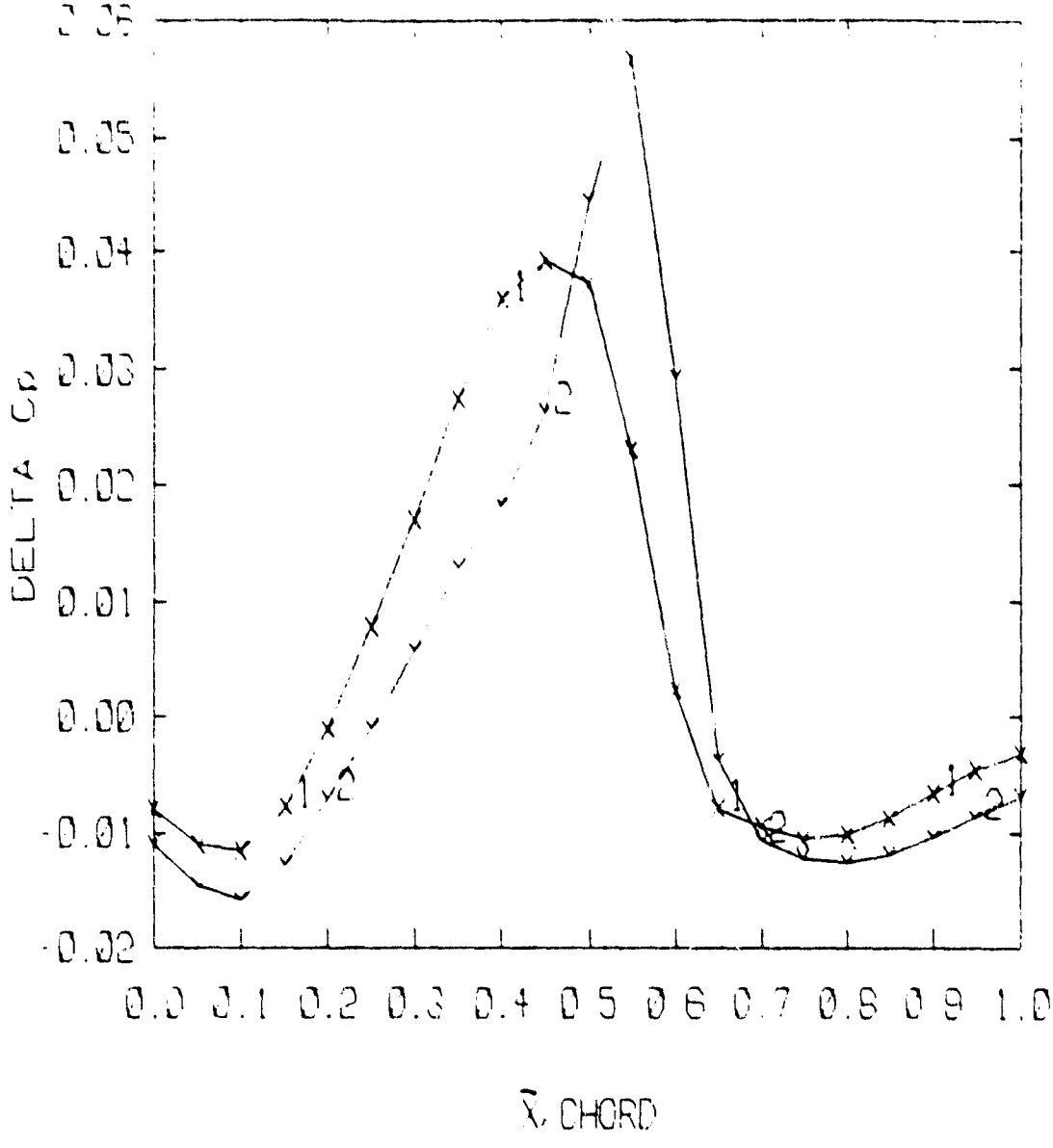


d) Perturbation 4

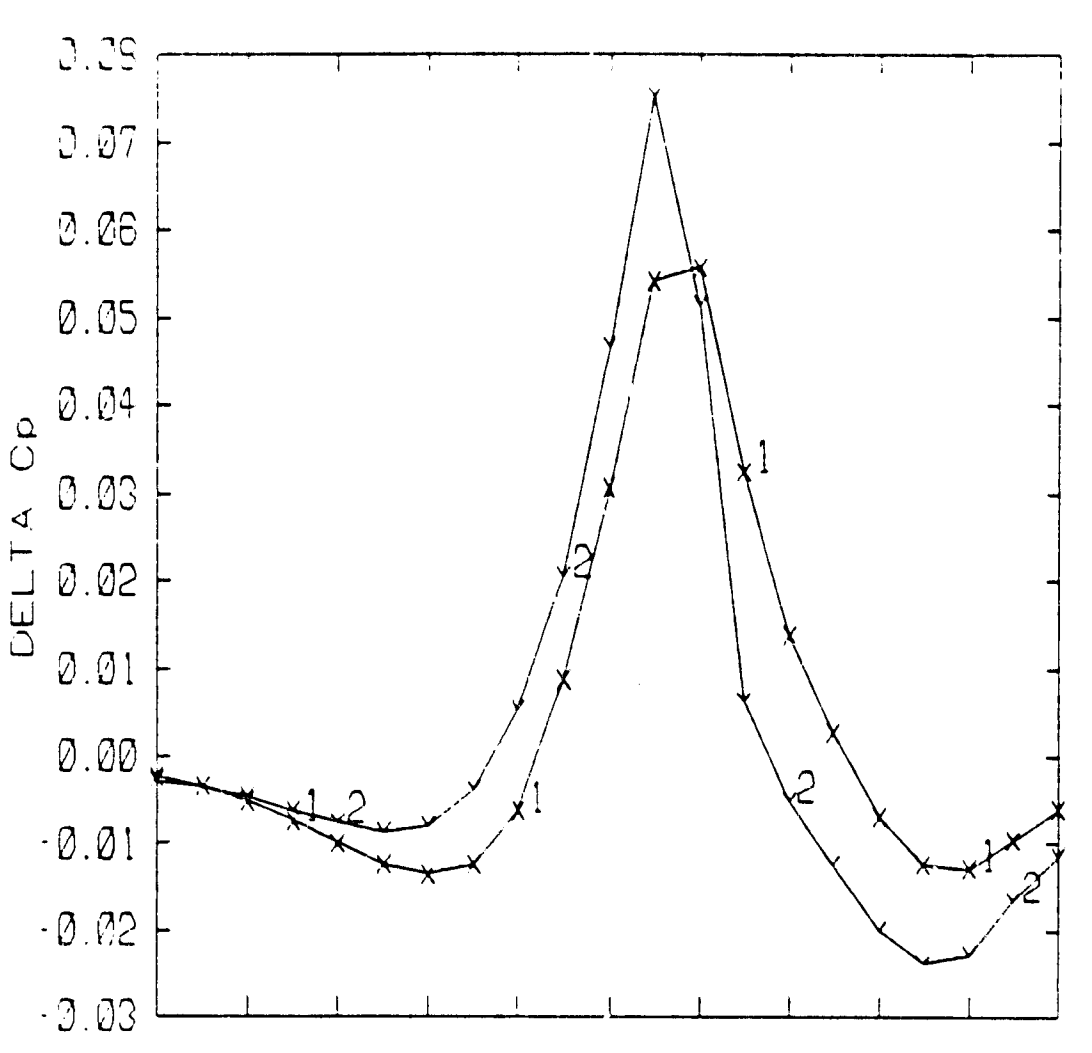


a) Perturbation 1

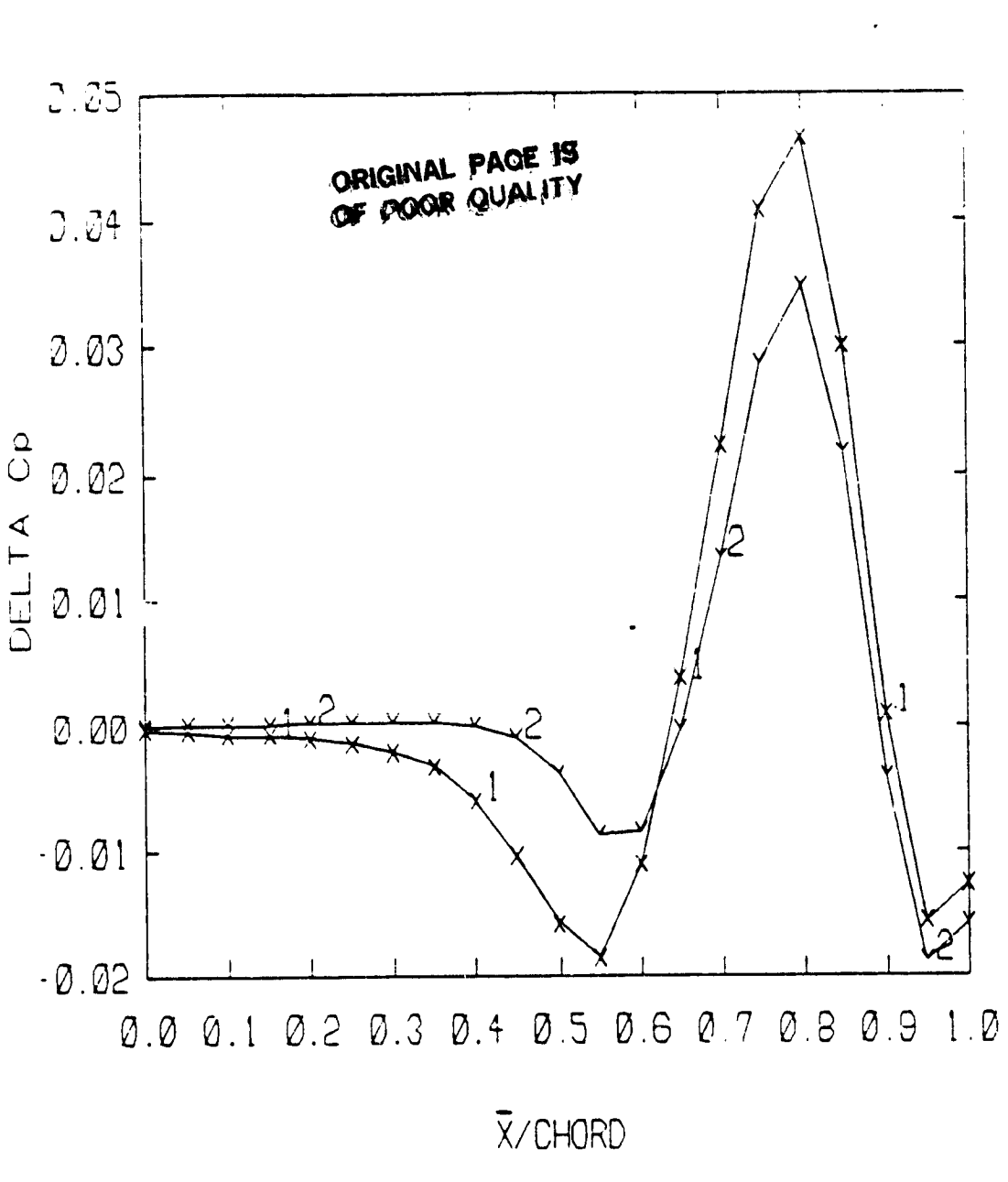
Figure 15. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Perturbation Transverse Velocity Added to Base Velocity at Top of Reduced Domain. M = 0.78, (t/c) =0.1



b) Perturbation 2



c) Perturbation 3



d) Perturbation 4

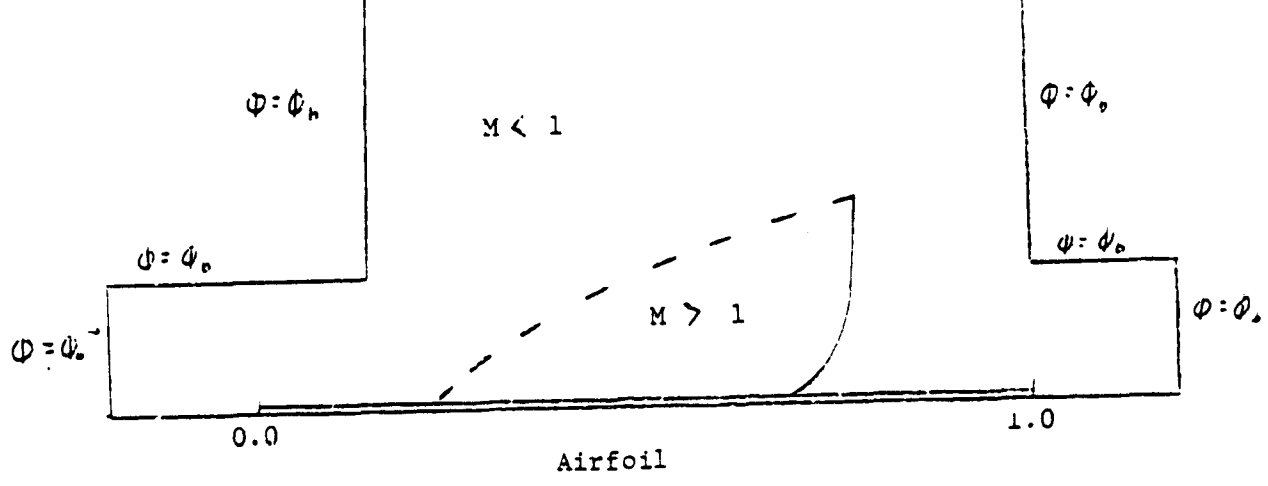
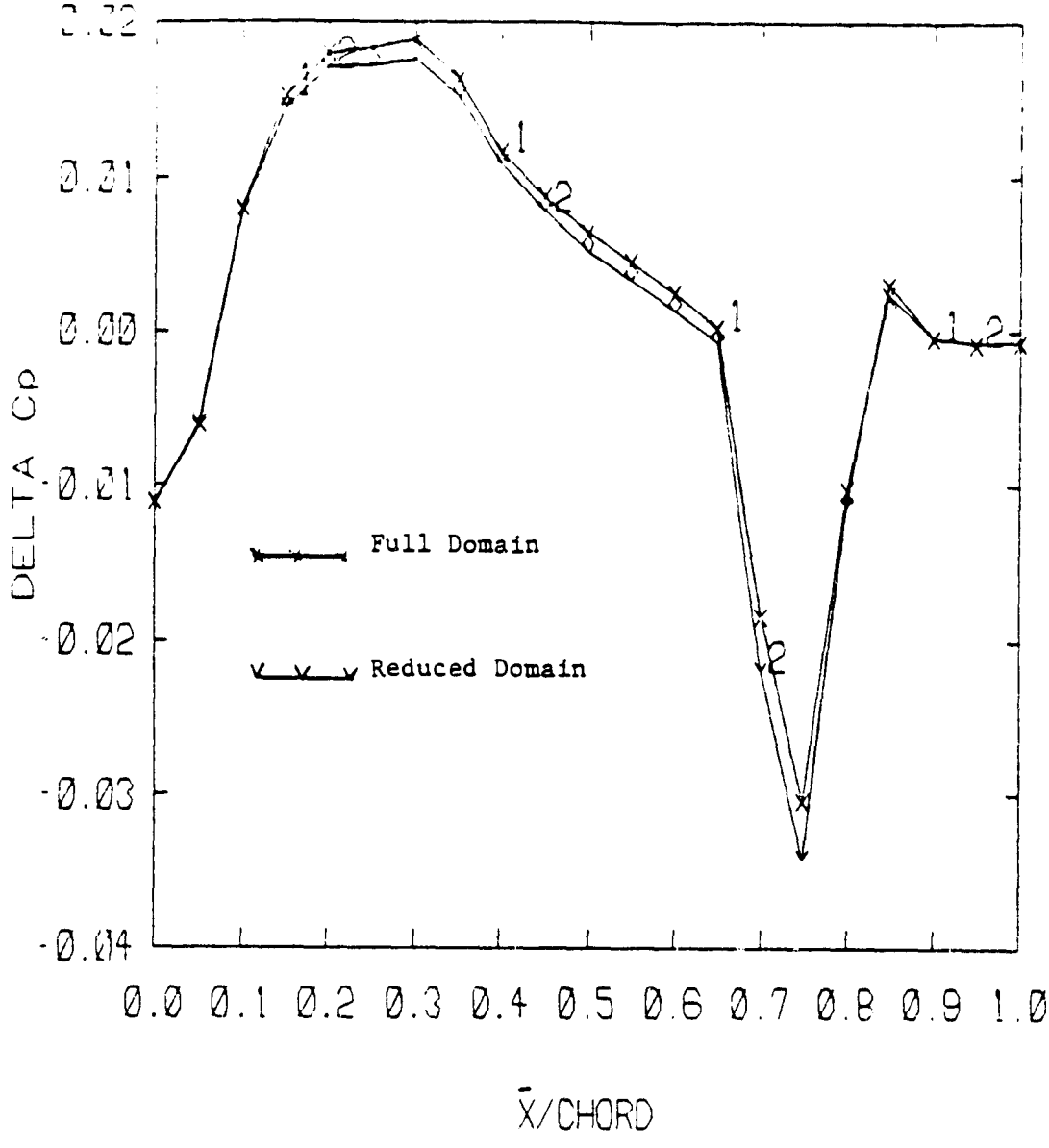
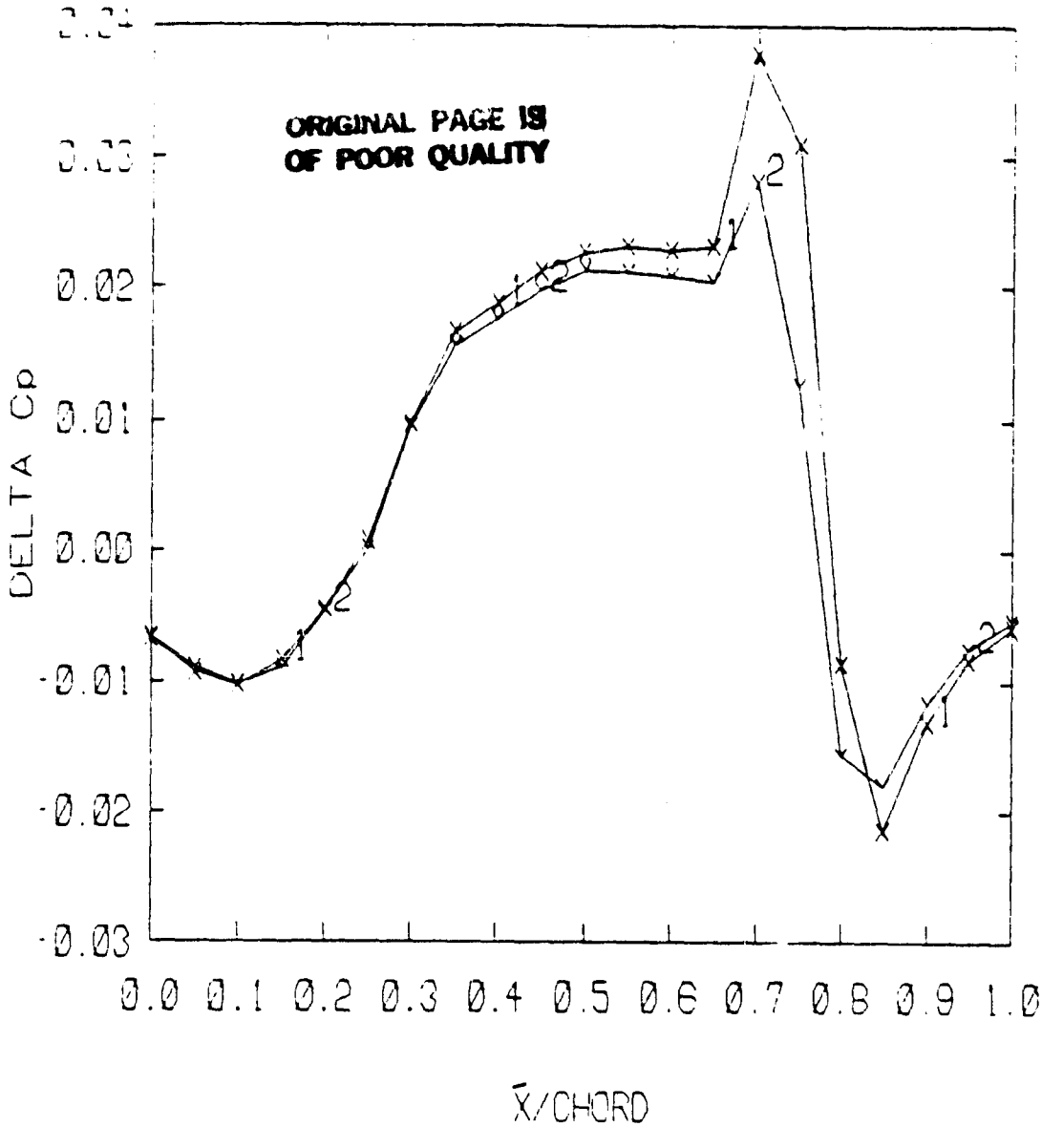


Figure 16. - Typical Notch Shaped Reduced Domain

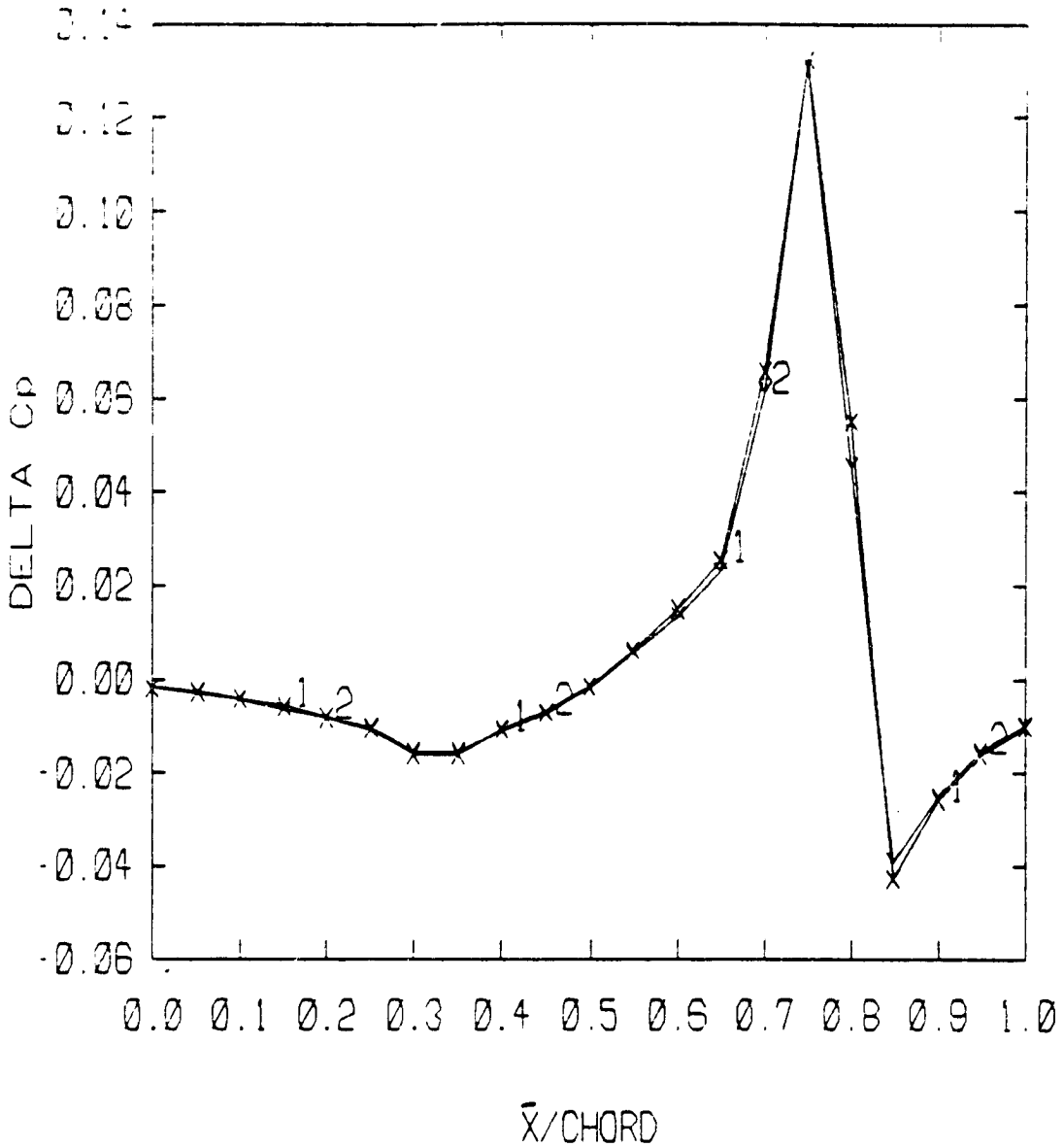


a) Perturbation 1

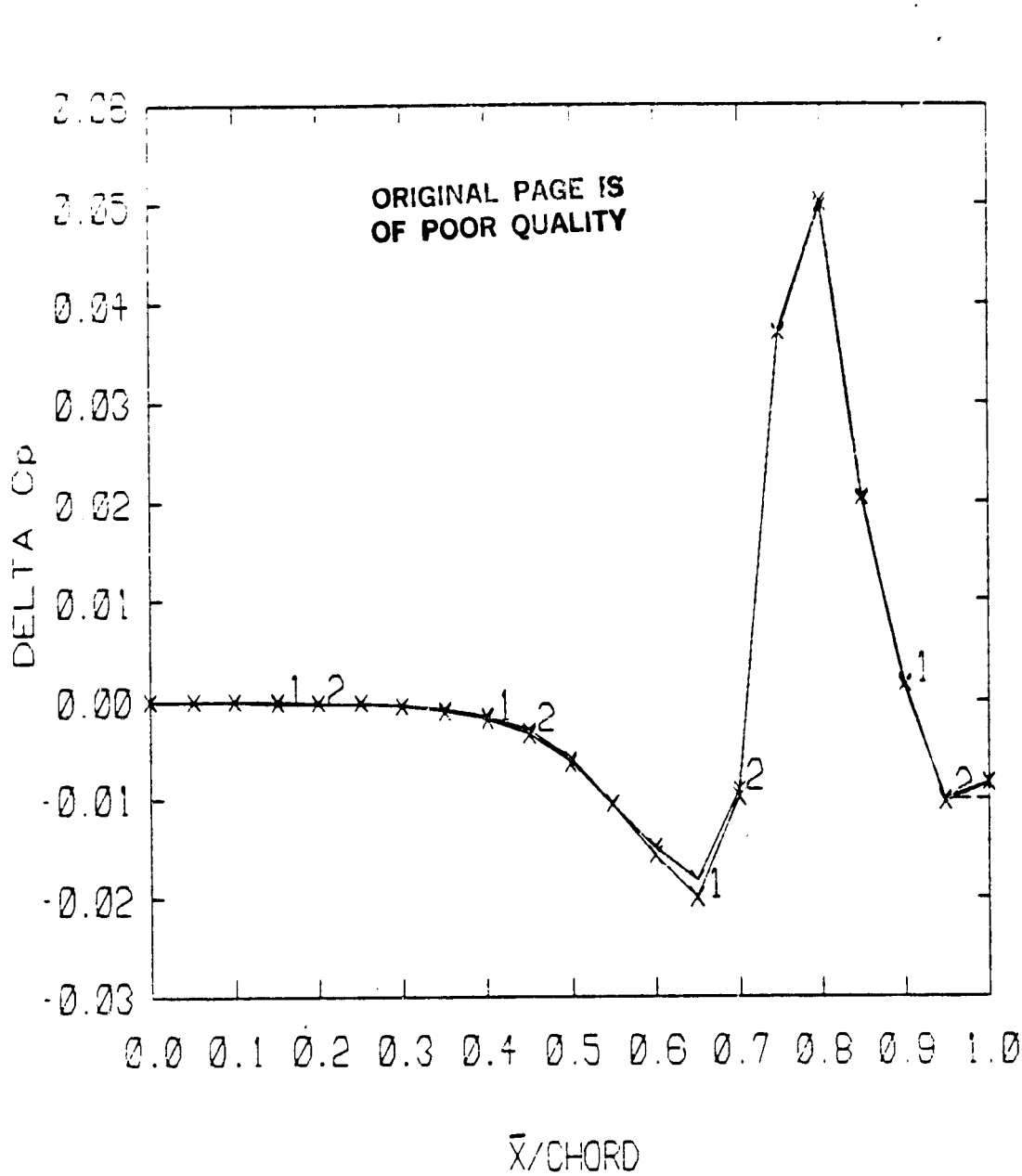
Figure 17. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Base Potential Specified at Edge of Reduced Domain. M = 0.82, (t/c) = 0.1



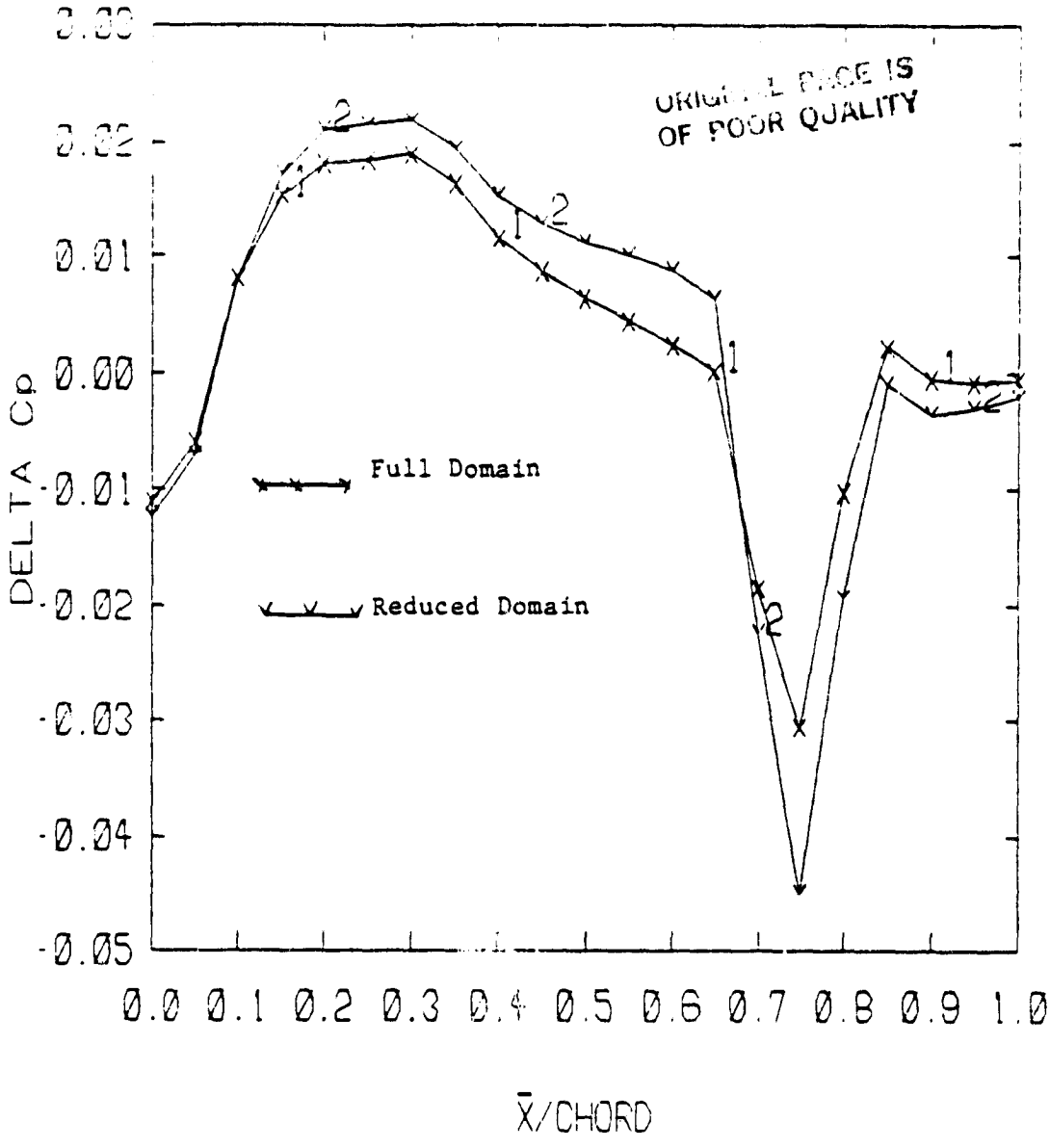
b) Perturbation 2



c) Perturbation 3

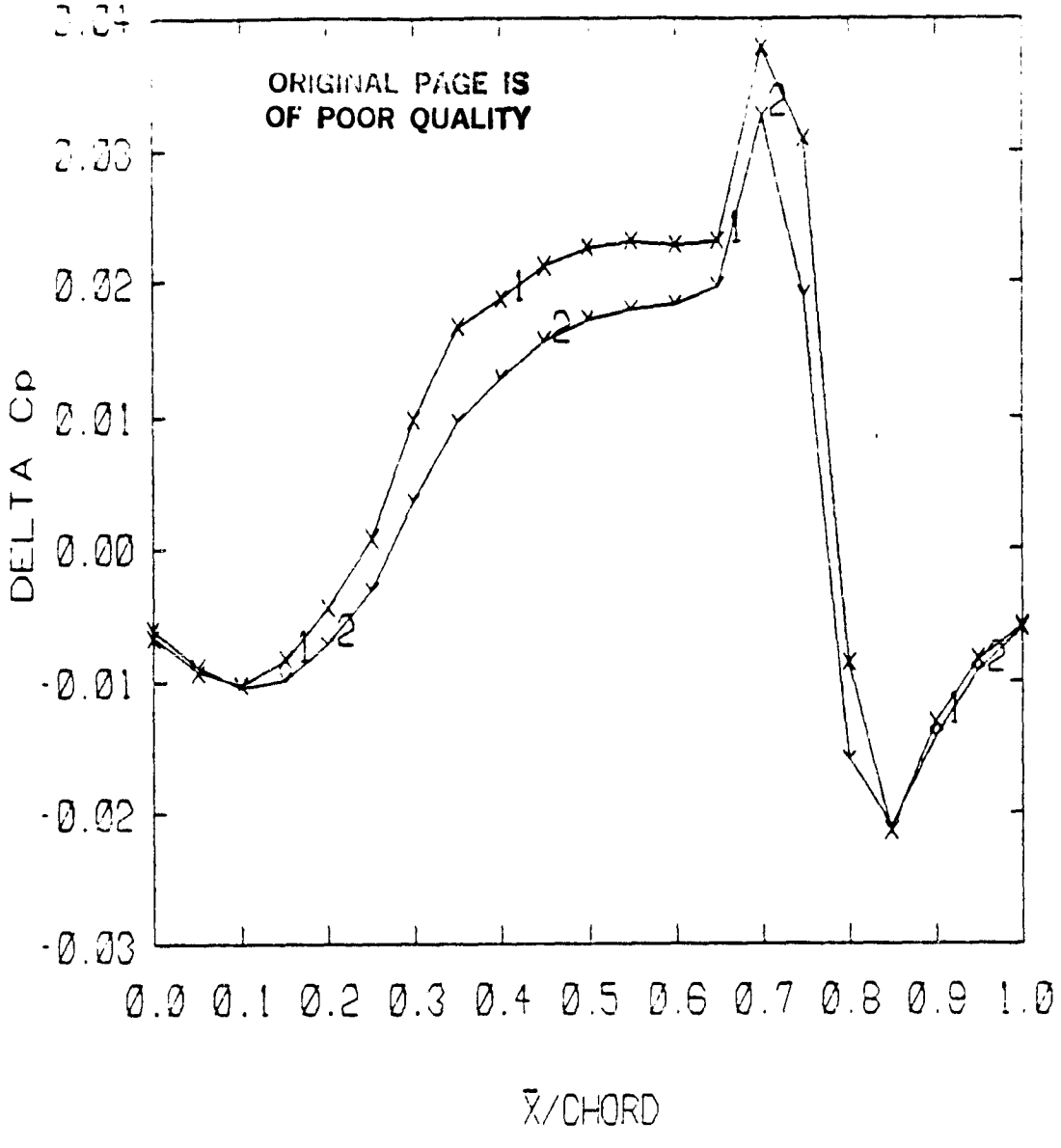


d) Perturbation 4

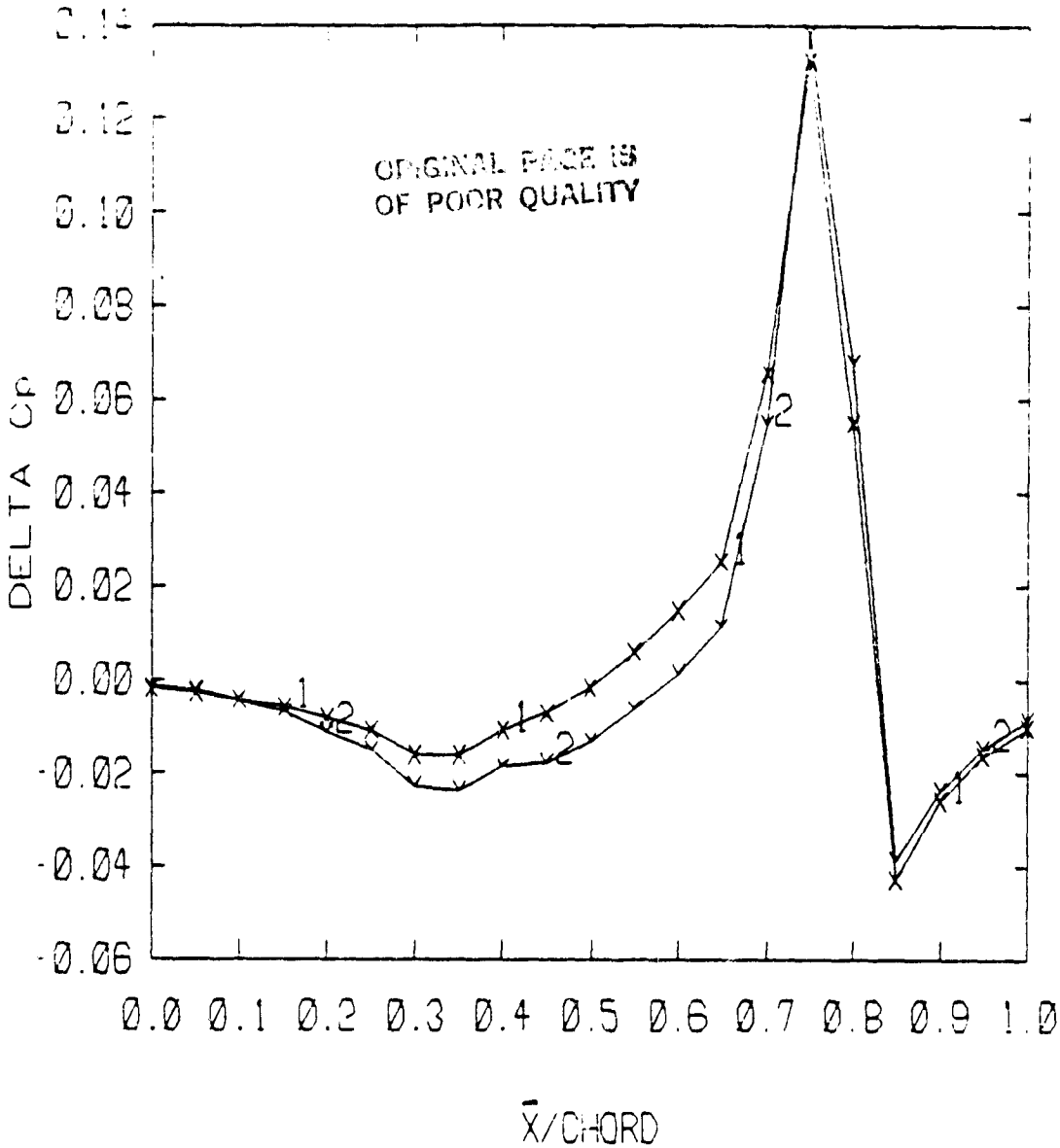


a) Perturbation 1

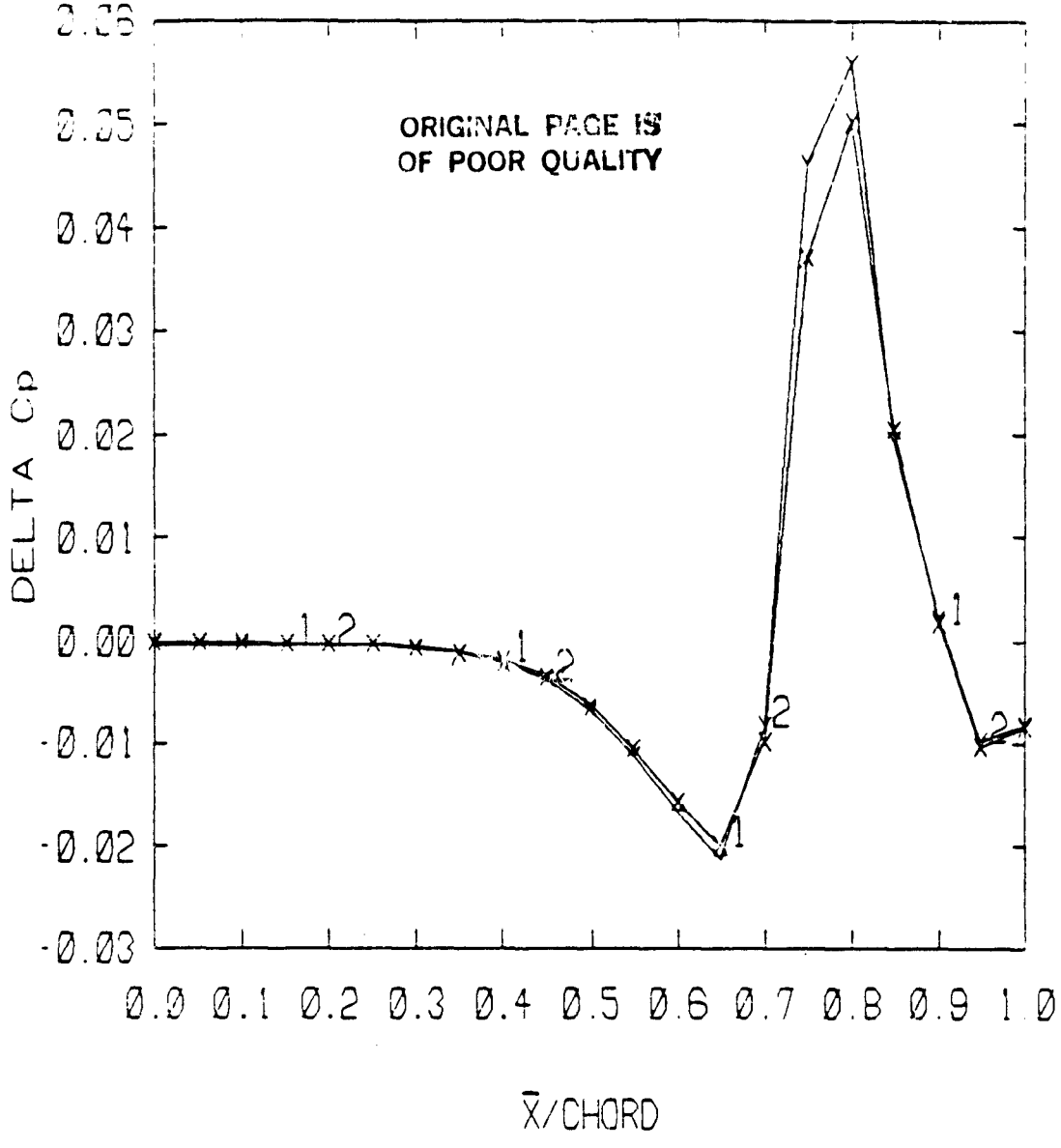
Figure 18. - Perturbation in Cp as Computed on Full Domain and Reduced Domain. Base Transverse Velocity Specified at Top of Reduced Domain. M = 0.82, (t/c) = 0.1



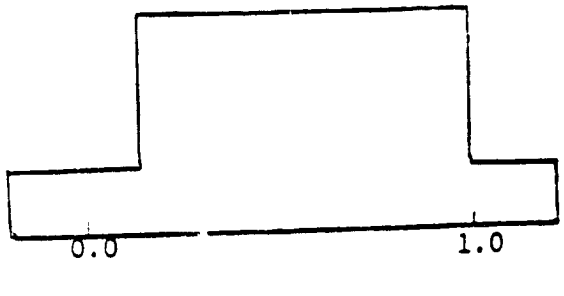
b) Perturbation 2



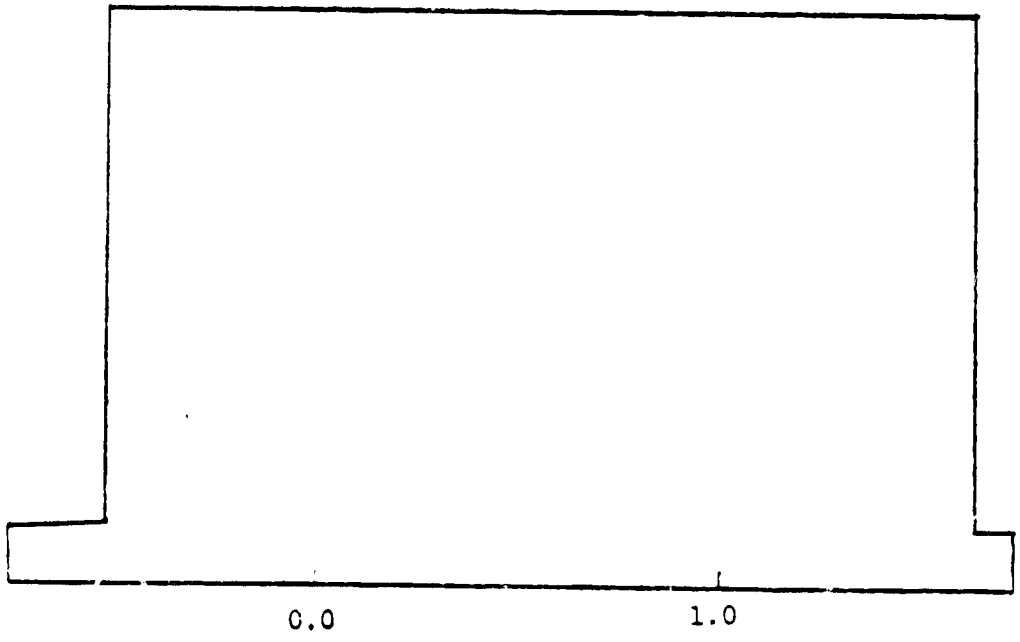
c) Perturbation 3



d) Perturbation 4



a) Neumann Boundary Condition



b) Dirichlet Boundary Condition

Figure 19. - Comparison of Domain Sizes Used

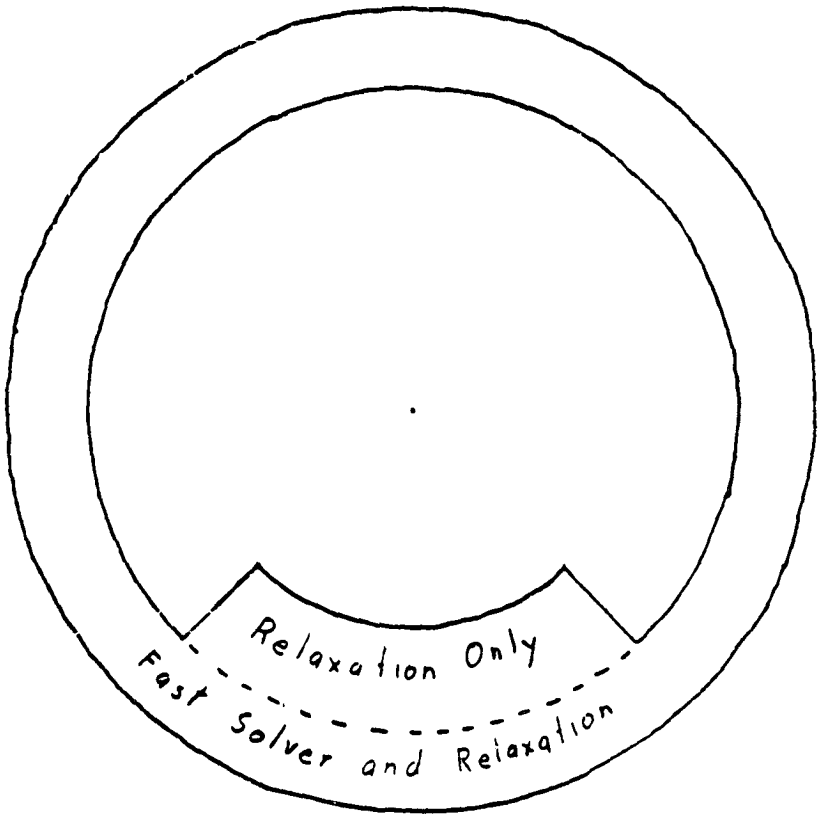
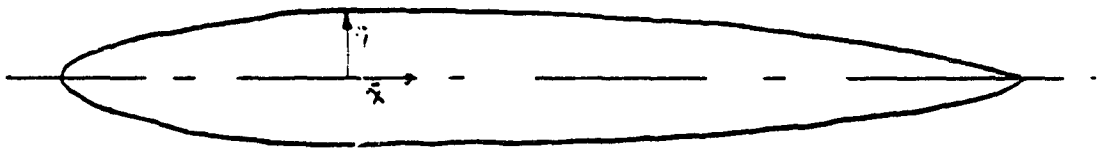


Figure 20. - Reduced Domain Used in Solving Full Potential Equation

C-2



$$\bar{y}/c = \left[(.07)^2\left(1 - \frac{(\bar{x}/c - .5)^2}{.5^2}\right)\right]^{1/2} \qquad \bar{x}/c \leq .5 \quad \text{(ellipse)}$$

$$\bar{y}/c = .28\left(\bar{x}/c\,(1 - \bar{x}/c)\right) \qquad \bar{x}/c > .5 \quad \text{(parabolic arc)}$$

Figure 21. - Airfoil Shape Tested

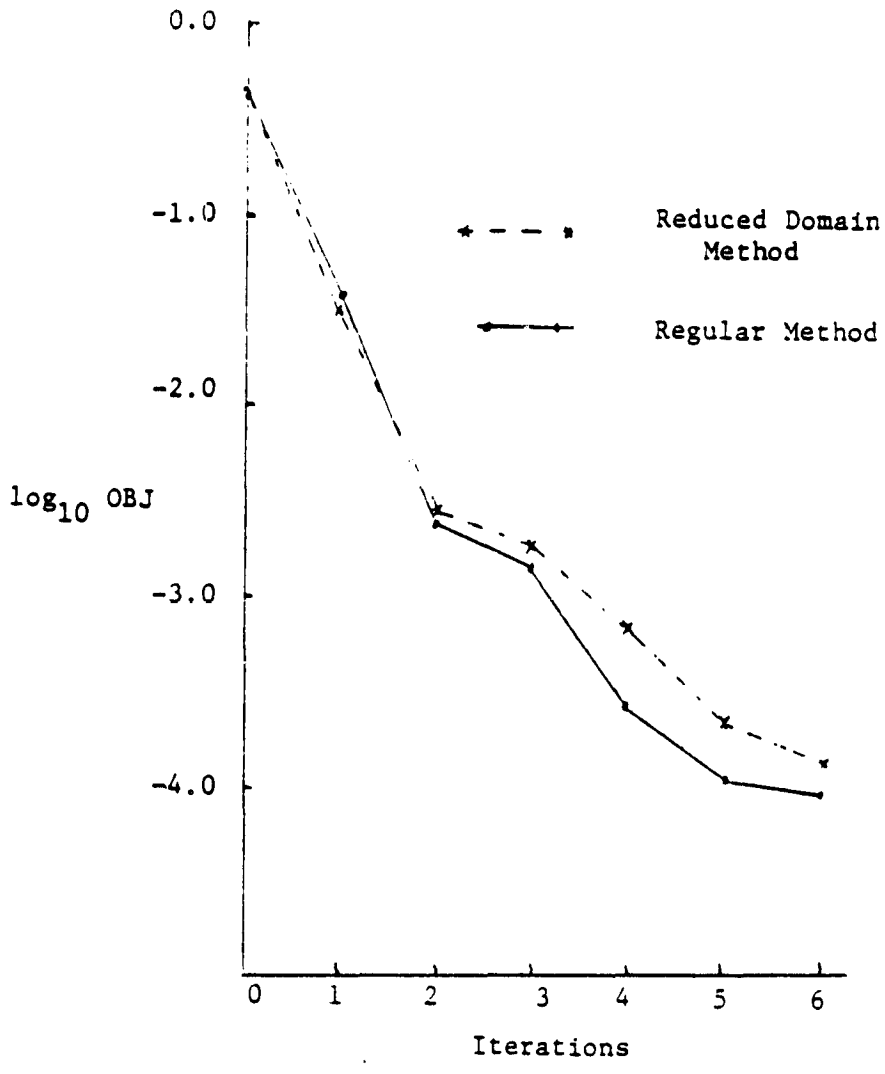


Figure 22. - Iteration History of Objective Function

APPENDIX. METHOD OF SOLUTION OF THE SMALL DISTURBANCE TRANSONIC POTENTIAL EQUATION

Consider the similarity form of the small disturbance transonic potential equation

$$\frac{\partial}{\partial x}\left(K\Phi_x - \frac{(\gamma+1)}{2}\Phi_x^2\right) + \frac{\partial}{\partial \tilde{y}}\left(\Phi_{\tilde{y}}\right) = 0 \qquad (A1)$$

where

$$K = (1-M_\infty^2)/M_\infty \delta^{2/3}$$

and

$$\tilde{y} = \delta^{1/3} M_\infty y$$

The pressure coefficient is ther given by

$$C_p = -2\Phi_x\left(\delta^{2/3}/M_\infty^{1/4}\right) \qquad (A2)$$

If the profile of the airfoil is given by

$$y = \delta f(x) \qquad (A3)$$

then the Neumann boundary condition, transferred to the axis is

$$\Phi_{\tilde{y}} = \frac{\partial f}{\partial x} \qquad (A4)$$

In the far field the airfoil is treated as a doublet. This gives for the potential in the far field

$$\Phi(x,\tilde{y}) \cong \left(\frac{1}{2\pi K}\right)\left(D x/(x^2+K\tilde{y}^2)\right) \qquad (A5)$$

where D = doublet stength

$$= 2\int_0^1 f(\xi)\,d\xi + \frac{\gamma+1}{2}\iint_{-\infty}^{\infty} \Phi_x^2\,d\xi\,d\eta \qquad (A6)$$

The numerical method used to solve this equation proceeds as follows. Let $p_{i,j}$ be a central difference approimation to the x derivatives

$$p_{i,j} = A_{i,j}\left[\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{\Delta x^2}\right] \qquad (A7)$$

where

$$A_{i,j} = K - (\gamma+1)\left[\frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}\right]$$

and let $q_{i,j}$ be a central difference approximation to the y derivatives

$$q_{i,j} = \frac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{\Delta \bar{y}^2} \qquad (A8)$$

Define the following switching function

$$\mu_{i,j} = 0 \quad \text{if} \quad A_{i,j} > 0 \quad \text{(subsonic)} \qquad (A9a)$$

$$\mu_{i,j} = 1 \quad \text{if} \quad A_{i,j} < 0 \quad \text{(supersonic)} \qquad (A9b)$$

Then a fully conservative finite difference scheme to solve the small disturbance potential equa is

$$p_{i,j} + q_{i,j} - \mu_{i,j}\, p_{i,j} + \mu_{i-1,j}\, p_{i-1,j} = 0 \qquad (A10)$$

Details of this method can be found in references 14,16, and 17.